

Avaneya Crew Handbook

A comprehensive guide for all of our bold contributors.

The Avaneya Crew

last updated 16 May 2011.

A comprehensive guide for all of our bold contributors.

Copyright © 2010, 2011 Kshatra Corp.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in [Section B.2 \[License GFDL\]](#), page 74.

Short Contents

1	Preface	1
2	The Game	2
3	Understanding Avaneya	4
4	Timeline	8
5	The Characters	13
6	The Story	14
7	The Game World	15
8	Gameplay	16
9	Selected Technology	17
10	Information For Everyone	21
11	Information For Engineers	22
12	The Project Crew	50
13	Licensing Rationale	59
A	Glossary	61
B	Licenses	63
	Index	87

Table of Contents

1	Preface	1
2	The Game	2
2.1	Why?	2
2.2	Classification	2
2.3	Who Is It For?	3
2.4	Who Is It Not For?	3
3	Understanding Avaneya	4
3.1	Resources For Everyone	4
3.2	Resources For Artists	5
3.3	Resources For Engineers	6
4	Timeline	8
5	The Characters	13
5.1	Arda Baştürk	13
6	The Story	14
6.1	Background	14
6.2	Storyboard	14
6.2.1	Opening Cinematic Sequence	14
7	The Game World	15
7.1	Time	15
8	Gameplay	16
9	Selected Technology	17
9.1	Audio	17
9.2	Developer Tools	17
9.3	Graphics	18
9.4	Input	19
9.5	Networking	19
9.6	Operating System	19
9.7	Physics	19
9.8	Scripting	19
10	Information For Everyone	21
10.1	Socioeconomic Modelling	21

11	Information For Engineers	22
11.1	AresPackages	22
11.1.1	Purpose	22
11.1.2	Usage	22
11.1.3	Design	22
11.2	Coding Standards	24
11.2.1	Legal	24
11.2.2	Mathematical	24
11.2.3	Formatting	24
11.2.4	Naming	28
11.2.5	Structure	29
11.2.6	Comments	29
11.2.7	Developer Tools	31
11.3	Architecture	31
11.3.1	Artificial Intelligence	32
11.3.2	Audio	33
11.3.3	Configuration	34
11.3.4	Console	35
11.3.5	Design Patterns	36
11.3.6	Engine	37
11.3.7	Events	38
11.3.8	File System	40
11.3.9	Goocy	41
11.3.10	Human Interface	42
11.3.11	Logging & Error Control	43
11.3.12	Mathematical	45
11.3.13	Miscellaneous	46
11.3.14	Physics	47
11.3.15	Resources	48
11.3.16	Scripting	49
12	The Project Crew	50
12.1	If You Are New	50
12.2	Specialities	50
12.3	Avaneya Code of Conduct	53
12.4	Current Members	54
12.5	Communication and Coordination	54
12.5.1	Mailing Lists	54
12.5.2	Internet Relay Chat (IRC)	56
12.5.3	Launchpad	56
12.5.3.1	Bug Tracker	56
12.5.3.2	Bazaar Revision Control	56
12.5.3.3	Blueprints	57
13	Licensing Rationale	59
Appendix A	Glossary	61

Appendix B	Licenses	63
B.1	GNU General Public License	63
B.2	GNU Free Documentation License	74
B.3	Creative Commons Attribution-NonCommercial-ShareAlike ...	81
Index		87

1 Preface

The Avaneya Crew Handbook has several purposes:

1. to define the Avaneya project as clearly as possible so that everyone involved understands it;
2. to provide this project definition in a single canonical location, which is useful when hacking in remote locations with unstable internet access;
3. and lastly, to save those already preoccupied with the project from having to explain the things which are redundant with this handbook.¹

This book is probably not as useful to normal users as it is to contributors. It may even spoil elements of the game for the former. On the other hand, normal users are just as likely to enrich the game through constructive feedback. Constructive feedback is best buttressed when coming from an understanding of the project.

¹ It is usually a good idea to ask questions. But as we say in the hacker community, read the documentation first.

2 The Game

2.1 Why?

Too often, people have come out of experiencing a great dystopian science fiction novel or film, only to say to themselves, Thank goodness we don't live in *that* world.

The problem here is that dystopian science fiction is among the most honest kind of story telling when examined in the context of history. But unless people can see the pedagogical relevance, analogies will remain vague and metaphors cryptic.

But good science fiction is only *partly* fiction because it is the *science* portion that attracts people to it in the first place. Science, by its nature, is a thinking and knowledge oriented enterprise. Science is the study of reality, and yet, fiction is outside of reality. So science fiction is a mediator between the two, often acting as an *avant-garde* for both.

This can set the bar high for what its patrons expect and requires designers to pay a great deal of attention to detail. A certain degree of creative license is expected, but people still prefer it to be as consistent with what we already know to be true, or what is at least reasonably plausible.

Another important reason why this project is necessary is that there is very little, if any, *free*, commercial, games for the GNU operating system. When people use GNU, they are treated as second class citizens in many respects - not least of which is the availability of good games. Not only are there very few higher production titles that are available for GNU, they are usually proprietary, and even then, generally bad ports using deprecated APIs, poorly packaged,¹ and integrate horribly into the user's desktop - ignoring the usual human factors community driven conventions.

In some sense, Avaneya is part of the effort to complete the GNU operating system. This is because "*a complete system needs games too*", once wrote Richard Stallman.²

But in terms of the games' subject matters themselves, they tend to appeal more to the mainstream proprietary user. They do not share the *software libre* community's values of freedom. To them, conscious computing is irrelevant since machines are just tools existing outside of a social context.

Worded differently, the choice between using free and proprietary software is not in the same class of arbitrary choices, like chocolate versus vanilla. Users of free software are qualitatively different from proprietary users. Therefore, a game for that audience should reflect this.

2.2 Classification

People have struggled in the past to classify Avaneya. It is what it is, but the closest traditional categories that form a subset of it are the traditional city builder and management simulations and the real time strategy.

¹ That is, if they even bother to use the platform's native distribution's package manager at all.

² <http://www.gnu.org/gnu/linux-and-gnu.html>

2.3 Who Is It For?

The game so far has attracted a fairly large base of followers. From what can be observed at this time, the game appears to appeal to those with an interest in:

- *challenging the consensus of reality*
- *software libre*
- *a social conscience*
- *science fiction*
- *the interconnectedness of everything*

The game may take place in the future, but it deals with current problems. The best way to get an idea of the intended audience is to quickly see [Chapter 3 \[Understanding Avaneya\]](#), [page 4](#) and to ask yourself the type of audience that those resources would probably resonate with.

2.4 Who Is It Not For?

Avaneya is a *sui generis*.³ It is not like other games, and thus it is not for all people. Those with a brief attention span, believe that things originate in cans with little appreciation for process, and accidentalists will probably not enjoy this game. There are already many such games that appeal to that type of audience, so that need not be our aim here.

This game will challenge you to think, and possibly even offend you. It challenges the consensus of reality, and therefore, potentially, your world view. Consequently, some have accused Avaneya of being a vehicle for culture jamming and political commentary. This project is shamelessly guilty as charged—like the newspapers, film, television, games, and other mainstream media that saturate us.

The only difference is that, unlike those mediums, the very presence of a normative bias in Avaneya is not subject to dispute and is self evident. Other mediums sometimes pretend to not have one. In any case, you would be very hard pressed to try to find any classical work of science fiction, or really any kind of fiction for that matter, that did not. Moreover, that in itself is not necessarily a bad thing.

³ "Literally meaning of its own kind / genus or unique in its characteristics. The expression is often used in analytic philosophy to indicate an idea, an entity, or a reality which cannot be included in a wider concept," (Wikipedia).

3 Understanding Avaneya

A great way to better your understanding of this project is to experience some of the relevant learning resources. You should be able to find all of these resources available either in digital format online, or in physical format at most major book stores. Most of them are not difficult to find.

3.1 Resources For Everyone

These resources are useful for every kind of contributor, from researchers and writers to modellers and engineers.

You might consider reading some of these resources in order to enrich your understanding of the game, whose fabric is drawn from *at least* all of them. The more background knowledge we have, the greater the quality of the game. The same goes for really any kind of complex simulation, and not just Avaneya.

- *Blue Mars: Mars Trilogy Bk. 3*
Kim Stanley Robinson, Voyager, ISBN 9780586213919, 1997.
- *Death by Government*
R. J. Rummel, Transaction Publishers, ISBN 9781560009276, 1997.
- *Debunking 9/11 Debunking*
David Ray Griffin, Olive Branch Pr, ISBN 9781566566865, 2007.
- *Free Software, Free Society: Selected Essays of Richard M. Stallman*
Richard M. Stallman, Free Software Foundation, ISBN 9781882114986, 2002.
- *Green Mars: Mars Trilogy Bk. 2*
Kim Stanley Robinson, Voyager, ISBN 9780586213902, 1994.
- *How to Live on Mars: A Trusty Guidebook to Surviving and Thriving on the Red Planet*
Robert Zubrin, Three Rivers Press, ISBN 9780307407184, 2008.
- *Merchants of Doubt: How a Handful of Scientists Obscured the Truth on Issues from Tobacco Smoke to Global Warming*
Naomi Oreskes et al., Bloomsbury Press, ISBN 9781596916104, 2010.
- *Minutes of Proceedings and Evidence Respecting the Bank of Canada*
Standing Committee on Banking and Commerce, 1939, pp. 461-500.
- *Open Letter to Minister of Public Safety Regarding 9/11*
Kip Warner, 2011.
- *Red Mars: Mars Trilogy Bk. 1*
Kim Stanley Robinson, Collins, ISBN 9780586213896, 1993.

- *Statistics of Democide: Genocide and Mass Murder since 1900 (Macht Und Gesellschaft, Bd. 2)*
R. J. Rummel, Lit Verlag, ISBN 9783825840105, 1999.
- *Terraforming: The Creating of Habitable Worlds (Astronomers' Universe)*
Martin Beech, Springer, ISBN 9780387097954, 2009.
- *The Case Against Fluoride: How Hazardous Waste Ended Up in Our Drinking Water and the Bad Science and Powerful Politics That Keep It There*
Paul Connett et al., Chelsea Green Publishing, ISBN 9781603582872, 2010.
- *The Case for Mars: The Plan to Settle the Red Planet and Why We Must*
Robert Zubrin et al., Free Press, 1997.
- *The China Study: The Most Comprehensive Study of Nutrition Ever Conducted and the Startling Implications for Diet, Weight Loss and Long-term Health*
T. Colin Campbell, Benbella Books, 2006.
- *The Creature from Jekyll Island: A Second Look at the Federal Reserve*
G. Edward Griffin, American Media, ISBN 9780912986395, 2010.
- *Zeitgeist: The Movie*
Peter Joseph, 122 minutes, GMP LLC, 2007.
- *Zeitgeist: Addendum*
Peter Joseph, 123 minutes, GMP LLC, 2008.
- *Zeitgeist: Moving Forward*
Peter Joseph, 161 minutes, GMP LLC, 2010.

3.2 Resources For Artists

These resources are useful for different kinds of artists, such as 2D artists, audio engineers, cinematic artists, modellers, musicians, and so on.

- *3D Computer Graphics, Second Edition*¹
Andrew Glassner, Green Editorial, ISBN 9781558213050, 1994.
- *Beginning GIMP: From Novice to Professional*
Akkana Peck, Apress, ISBN 9781430210702, 2009.

¹ This book is listed as a resource for artists because it is a wonderful and concise introduction to the theory of computer graphics, but for non-programmers and non-mathematicians. It is a classic, though long out of print. Nevertheless, you can probably find used copies for sale.

- *Blender 2.5 Materials and Textures Cookbook*
Colin Litster, Packt Publishing, ISBN 9781849512886, 2011.
- *Blender 3D 2.49 Incredible Machines*
Allan Brito, Packt Publishing, ISBN 9781847197467, 2009.
- *Blender 3D Architecture, Buildings, and Scenery: Create photorealistic 3D architectural visualizations of buildings, interiors, and environmental scenery*
Allan Brito, Packt Publishing, ISBN 9781847193674, 2008.
- *Blender Studio Projects: Digital Movie-Making*
Tony Mullen & Claudio Andaur, Sybex, ISBN 9780470543139, 2010.
- *Introducing Character Animation with Blender*
Tony Mullen, Sybex, ISBN 9780470102602, 2007.
- *The Essential Blender: Guide to 3D Creation with the Open Source Suite Blender*
Roland Hess, No Starch Press, ISBN 9781593271664, 2007.

3.3 Resources For Engineers

These resources are useful mostly for engineers and other very technical work.

- *Autotools: A Practioner's Guide to GNU Autoconf, Automake, and Libtool*
John Calcote, No Starch Press, ISBN 9781593272067, 2010.
- *Beginning Game Audio Programming*
Mason McCuskey et al., Premier-Trade, ISBN 9781592000296, 2003.
- *Beginning iPhone Games Development*
PJ Cabrera et al., Apress, ISBN 9781430225997, 2010.
- *Design Patterns: Elements of Reusable Object-Oriented Software (Addison-Wesley Professional Computing Series)*
Gamma et al., Addison-Wesley Professional, Hardcover, ISBN 9780201633610, 1994.
- *Game Engine Architecture*
Jason Gregory, A K Peters/CRC Press, ISBN 9781568814131, 2009.
- *Lua 5.1 Reference Manual*
Roberto Ierusalimsky et al., Lua.org, ISBN 9788590379836, 2006.
- *Lua Programming Gems*
Lua.org, ISBN 9788590379843, 2008.

- *OpenGL Library (5th Edition)*
Dave Shreiner et al., Addison-Wesley Professional, ISBN 9780321637642, 2009.
- *Pro OGRE 3D Programming*
Gregory Junker, Apress, ISBN 9781590597101, 2006.
- *Programming in Lua, Second Edition*
Roberto Ierusalimsky, Lua.org, ISBN 9788590379829, 2006.
- *Systems and Models. Complexity, Dynamics, Evolution, Sustainability*
Hartmut Bossel, BoD, ISBN 9783833481215, 2007.
- *System Zoo 1 Simulation Models - Elementary Systems, Physics, Engineering*
Hartmut Bossel, BoD, ISBN 9783833484223, 2007.
- *System Zoo 2 Simulation Models. Climate, Ecosystems, Resources*
Hartmut Bossel, BoD, ISBN 9783833484230, 2007.
- *System Zoo 3 Simulation Models. Economy, Society, Development*
Hartmut Bossel, BoD, ISBN 9783833484247, 2007.

4 Timeline

The world Avaneya takes place in is highly self-referential. A fictional timeline is positioned relative to a *year zero*, with dates preceding suffixed *B.R.*, before the republic; while those following are suffixed *A.R.*, after the republic. This is the method adopted by Arda and his settlers.

Note that a year is defined as a Martian year (MYr) and is not the same as a Terran year. You may want to refer to [Section 7.1 \[Game World Time\]](#), [page 15](#) for a more detailed explanation on how the settlers keep track of time on Mars.

The year zero's placement relative to the Gregorian calendar is deliberately never given. But inferentially, one can reason that it is proximate enough for a contemporary audience to still feel relevant, yet distant enough to afford some reasonable creative licenses.

- **12 Feb 75 B.R.** The last of the *International Space Station* crew return to Earth with the station subsequently decommissioned, deorbited, and incinerated during atmospheric entry.
- **26 June 42 B.R.** The bancor is declared the new global currency and implemented via the respective central banks of the North American Union, European Union, African Union, and Asian Union.
- **13 Dec 39 B.R.** Ramraj discovers a near-Earth, C-type Amor II asteroid, with a size comparable to Phobos from the Lincoln Near-Earth Asteroid Research laboratory, Socorro, New Mexico, North American Union. The discovery is rapidly assigned the designation *52048 Varuna* by the *International Astronomical Union*. His findings are detailed in the *Minor Planet Circular* where he calculates a near collision trajectory with Mars.
- **13 Dec 37 B.R.** Wikileaks publishes a leaked list of 163 purported attendees of the annual Bilderberg conference held three days prior at the Hotel de Crillon, Paris, European Union. Dr. Samuel Lieberman, director of the National Institute of Standards and Technology, and Adriaan Janssen, the Secretary-General of the United Nations, are among those listed.
- **15 Sep 35 B.R.** The United Nations holds the International Mars Summit in Geneva, Switzerland, with the purpose of discussing the options on the table for the first manned mission to Mars. The attendees represent the African Union, North American Union, European Union, Asian Union, along with thousands of scientists, engineers, philosophers, and other scholars from around the world. The minimalistic Mars Direct approach of travelling light, living off of the land, and using indigenous materials to produce the fuel necessary for the journey home is a central theme.
- **28 Sep 35 B.R.** Jacob Kissinger, president of the *Council of Foreign Relations* holds a symposium at the Harold I. Pratt House in New York City, North American Union.
- **3 Oct 35 B.R.** The *Trilateral Commission* holds its biannual meeting several months

in advance, Tokyo, Asian Union.

- **12 Oct 35 B.R.** The International Mars Summit concludes publishing its findings in the *Mars Report*: "*Although the Mars Direct approach is totally feasible, the bar can and should be pushed. The summit determined that it is best to go, not merely for the sake of demonstrating that a safe return is possible, but to establish a permanent settlement on Mars. The purpose being to bootstrap the first non-terrestrial expansion of human civilization for the benefit of all mankind.*"

A preliminary draft of the *Mars Treaty* is produced using a revised *Outer Space Treaty* of 1967 A.D. as a basis for a subset of it. The treaty provides the contractual basis for all signatories, outlining the roadmap, responsibilities, and party resource allocations necessary to serve as a vehicle for the terraformation and first manned mission to Mars operating under a United Nations mandate. The treaty also provides an agreement governing the activities of states on Mars and its two moons, Phobos and Deimos.

The terraformation strategy is to be initiated by intercepting 52048 *Varuna* and deflecting it to directly impact Mars using a nuclear warhead. Being a C-type asteroid, it contains high levels of hydrated (water-containing) minerals which, when released into the atmosphere, will greatly increase cloud condensation nuclei. Data was presented at the summit predicting a dramatic [increase / decreasing] in albedo, the proportion of sunlight reflected back into space. This would cause a cascading series of reactions, increasing the mean surface temperature and atmospheric pressure.

This international project implicated by the treaty is termed the *Avaneya Initiative*.

- **22 Oct 34 B.R.** *United Nations General Assembly Resolution 5571* is adopted making the Mars Treaty ratified. The resolution augments *Chapter III* of the *United Nations Charter* to beget its seventh principle organ, the *Space Agency* (UNSA). It is bestowed with a mandate enacting the treaty, and by implication, the *Avaneya Initiative*.

The United Nations Office for Outer Space Affairs (UNOOSA) is dissolved. Its former responsibilities are transferred to UNSA under its new mandate.

The resolution also requires members to accede the North American Union Aeronautical and Space Agency (NASA), European Space Agency (ESA), Asian Aerospace Exploration Agency (AAXA), and all other national member space agencies under centralized UNSA administration. This is granted under the justification that space exploration and settlement ought to be an endeavour for all mankind, best realized through the aggregation of resources.

Lieberman is nominated to the Office of the President of the UNSA, resigning his position as director for the National Institute of Standards and Technology.

- **3 Aug 31 B.R.** The UNSA's Office of the *Avaneya Initiative* disseminates more than ten-thousand requests for quotations.
- **17 Jan 30 B.R.** UNSA's Office of the *Avaneya Initiative* announces successful bids. The *Iterum Shipyard* contract is jointly awarded to Lockheed Martin-Boeing and Mercedes-Pratt & Whitney to build a station in low Earth orbit at a cost of 2.2 billion bancors. The station is to be used as a general purpose interplanetary space vehicle construction platform.

Lockheed Martin-Boeing is awarded a contract to provide three, leased, single-stage-to-orbit (SSTO), manned, re-usable, *VentureStar VII* suborbital spaceplanes. These spaceplanes provide UNSA with heavy lift workhorses for transporting personnel and materials to the Iterum Station. They require no external solid booster rockets or external fuel tank and each operate at less than $1/5^{th}$ the cost of the long retired Space Shuttle predecessor.

Mitsubishi-Saab's is awarded a contract to provide the *Avaneya*, constructed *in situ*, at the *Iterum Shipyard*, at a cost of 7.8 billion bancors.

Volvo-John Deere is awarded a contract to provide *Avaneya's* nuclear electric ion-drive propulsion system at a cost of 920 million bancors.

Soros-Murdoch Aerospace is awarded a contract to provide four *Mars Positioning System* satellites and the *Mars Enhanced Telecommunications Orbiter* carrying an optical relay at a cost 750 million bancors.

- **26 July 29 B.R.** Construction of the first completed UNSA funded training facilities in Antarctica and Huelva, Spain, are completed.
- **1 Sept 29 B.R.** Crew selection and training begins at training facilities.
- **B.R.** Crew selection is completed with a broad range of personnel including flight engineers, artificial intelligence specialists with knowledge engineers among them, cyberneticists, chemical and civil engineers, cold weather construction experts, mechanics, biogeochemists, geologists, areobotanists, one xenobiologist, and more. Arda Baştürk is named Mission Commander.
- **B.R.** *Mars Science Laboratory Curiosity XI*, an unmanned autonomous aerial vehicle, explores potential landing sites for settlement. The integrated on-board artificial intelligence is instructed to evaluate sites based on average available sunlight, water ice, mineralogy, surface geography, and other factors.
- **B.R.** *Internet Assigned Numbers Authority* allocates A001:CA7:3134::/48 IPv6 address block for Martian use.
- **B.R.** Last rocket leaves UNSA's Cape Canaveral launchpad carrying construction material for the *Iterum Shipyard*.
- **B.R.** *Avaneya* construction is completed *in situ* at the *Iterum Shipyard*, low Earth orbit. The station orbits Earth at an altitude of 340 km, travelling at a speed of 27,400 km/h, and taking one and a half hours to complete one orbital revolution.
- **B.R.** Launched from the Kennedy Space Center, Florida, an SSTO transporting the *Avaneya* crew dock with the *Iterum Shipyard*, low Earth orbit.
- **7 Jan 8 B.R.** *Avaneya* completes all system checks and disembarks *Iterum Shipyard* berth. It performs a delta-v from low Earth orbit into a type II Hohmann transfer orbital manoeuvre.

- **27 Aug 8 B.R.** *Avaneya* deploys four *Mars Positioning Satellites* (MPS) into medium Martian orbit.
- **28 Aug 8 B.R.** *Avaneya* aerobreaks into Martian orbit. Instrumentation subsystems update onboard areology database on detailed surface geography, weather dynamics, mineralogy, and more.
Mars Enhanced Telecommunications Orbiter is released and deploys itself into geostationary orbit.
- **29 Aug 8 B.R.** *Mars Enhanced Telecommunications Orbiter* successfully completes self diagnostics and registers itself into UNSA's *Interplanetary Internet* as an available communications node, giving rise to what becomes colloquially known as *Solnet*. Terrestrial downlink passes through a ground station at UNSA's Jet Propulsion Laboratory providing tier 1 network access.
Brokered by a satellite uplink with METO, the *Avaneya* communications team establish contact with Mission Control routed over Solnet. Solnet becomes the defacto standard for Earth-Mars intercommunication.
- **30 Aug 8 B.R.** Landing site selected. Cargo of mostly construction equipment, hydrogen fuel, water, and other provisions, are jettisoned and parachuted to surface with no material loss, save one asset due to an attitude control computer malfunction.
- **30 Aug 8 B.R.** All crew alight the *Avaneya*, boarding the *Manu* landing craft. The *Avaneya* tracks the critical manoeuvres of *Manu*'s entry, descent, and soft landing at the preselected drop site. The communications uplink with Mission Control is maintained throughout over a 4 minute delayed Solnet connection.
Crew perform a successful soft landing where they are met with a temperature of -70°C and an atmospheric pressure of 0.030 kPa. Time is local noon to maximize photovoltaic use.
Team briefings are conducted within *Manu* at the drop site. Concurrently, the recovery team are deployed to recover all assets.
The construction team begin excavation for anchoring and erecting temporary inflatable tents and greenhouses.
- **3 Sep 8 B.R.** Gas extractors are brought online and run at full capacity capturing liquid oxygen, liquid nitrogen, argon, and carbon dioxide.
Sebatier, RWGS, and methanol gas processors¹ create methane, oxygen, hydrogen, methanol, and aqua successfully.
- **14 Sep 8 B.R.** Recovery team's rovers directed by scouts on methanol motorbikes complete asset recovery of all undamaged parachuted cargo within a 92 kilometre radius from the crew landing site.
- **16 Sep 8 B.R.** Construction team bring backhoes, front loaders, bulldozers, tractors,

¹ See [Appendix A \[Glossary\]](#), page 61 for more information on the Sebatier and RWGS reactors.

graders, water ice processors, and dump trucks online.

On-board artificial intelligence and system firmware is upgraded from UNSA's Jet Propulsion Laboratory over Solnet.

Mining and excavation operations begin through a mixture of directly manned, remotely manned, and autonomous operation.

- **7 Sep 8 B.R.** Flight engineers issue instructions remotely to *Avaneya* to return to Iterum Shipyard from the surface.
- **[Year Zero] 3 May 0 A.R.** Major-General Arda Baştürk informs the Secretary-General of the United Nations via an internationally televised address of the passage of the Rubicon Act, initiating steps to secure the settlement's independence. The Act declares the colony an independent, sovereign, self-governed, constitutional republic, with a right to self determination. *Article IX* discharges all public debt held by the *International Monetary Fund* and *World Bank* effective immediately. The Bancor fiat currency is replaced with a rhodium² standard as legal tender within the republic.
- **4 May 0 A.R.** Major-General Arda Baştürk has a security detail escort United Nations envoy Jacob Kissinger to *EVA Bravo*, Earth-bound.
- **May 5, 0 A.R.** United Nations Security Council Resolution 12664 is adopted which "*strongly condemns the actions of the illegitimate government of X and calls upon all members to proactively prevent the regional instability and deteriorating situation of the X settlement*". The Security Council resolution recommends to the *United Nations Department of Peacekeeping Operations* (UNDPKO) that it place all four permanent standing *Rapid Reaction Force* battalions on high alert.
- **May 19, 0 A.R.** Selected Rapid Reaction Force personnel are assigned to UNSA training facilities in Antarctica and Huelva, Spain.
- **April 5, 1 A.R.** The *Yama*, a manned spacecraft, completes a type I Hohmann transfer orbital manoeuvre, aerobreaking into Martian geostationary orbit.
Yama carries a payload of remotely operated equipment destined for Phobos, as well as a single Rapid Reaction Force battalion representing the *United Nations Emergency Assistance Peacekeeping Force*. Its numbers are drawn principally from North American Union and European Union airborne light infantry units.
- **April 5, 1 A.R.** Yama's Commanding Officer Lieutenant-Colonel Dragov issues warning orders down the chain of command to prepare for insertion, geostationary orbit, Mars.

² See [Appendix A \[Glossary\]](#), page 61 for more information on rhodium.

5 The Characters

Avaneya's main characters aim to be rich in detail and multifaceted. Their character attributes are generally revealed through indirect means, such as body language, object symbolism, and dialogue. This may happen during a cinematic sequence, for instance.

5.1 Arda Baştürk

Arda was born 43 B.R., Istanbul, Turkey, to Dursun and Ayla Baştürk. His mother, Ayla, was a bank teller, and Dursun, an aeronautical engineer with the NAUSA Jet Propulsion Laboratory and later military intelligence officer.

Dursun had been active politically in opposing Turkey's membership in the European Union which happened shortly after Arda's birth. Several attempts were made on Dursun's life, culminating in a car bomb that left him paralysed before finally succumbing to his condition a year later.

Ayla was cautious in discussing the identity of the suspected assailants with Arda, as she did not want to risk putting her son in danger. Still, Arda sensed she had suspected that it had not been the work of marginalized extremists, but had been the work of elements within their own government on behalf of the Rothschilds.

As part of a working class family, with only his mother to support him, Arda had few opportunities that would not come without extending great effort on his part.

In an era defined by scarcity, they were of modest means and she felt it important to continue imparting Dursun's philosophy of a positive work ethic. With food shortages rampant, high unemployment, and riots on a weekly basis in Istanbul, she knew it was his only ticket out, if he should have any at all.

Arda was described in his youth as highly perceptive and intelligent, although generally reserved, and further, reserving his intellectual gifts for his principle love, science. With the exception of what little he knew of his father, he felt the political landscape was largely the realm of "*bumbling idiots*". This, combined with his privately held view that the world was irreversibly falling apart, he preferred to devote himself to what was immutable, studying the forces governing the world of natural phenomena.

Arda studied at Istanbul Technical University, made possible through a combination of scholarships and his salary as a part time army reserve officer cadet. He graduated with a degree in aeronautical engineering, 21 B.R., where he was then commissioned into the army as a lieutenant. He completed his doctorate in heliophysics two years later, with rumours he solved a central problem that plagued his thesis while deployed on a field exercise.

Although wanting to focus on his research, even with his qualifications, employment was scarce and the army reserves was one of the few opportunities available that was stable and left him with enough time of his own. He was well respected in the army as a natural born leader, though his privately held view of the institution was one of resentment, corruption, and misuse.

He later underwent commando training with the 3rd Commando Brigade, Siirt, before being posted to the Mountain and Commando Brigade, Hakkari. He rose through the ranks before acting as the unit's liaison officer, overseeing North American Union troops undergo training at their facilities.

6 The Story

6.1 Background

6.2 Storyboard

6.2.1 Opening Cinematic Sequence

7 The Game World

This chapter deals with some of the different aspects of the world the game takes place in, as one in that world itself might know it as.

7.1 Time

The colonists use a different calendar than on Earth for both practical and political reasons.

Practical, because the Gregorian calendar is useless in a world where the orbital period, seasons, and length of a solar day are different than on Earth. The calendar's months, for instance, are not meaningful in the absence of the natural cycles of the Earth's Moon.

Political, partly because the original Avaneya settlers were predominantly secular, the Gregorian calendar's year zero being incompatible with that; and partly that they desired a calendar which was meaningful in the context of *their own* collective memory.

But as the details go, a Martian hour is the same as a Terran hour, composed of 3600 Terran seconds.

A *sol* is a solar day.¹ This is the Martian analogue to a Terran day, but 2.7 % longer. The conversion of 1.027491 Terran days to Martian sols means that a sol is slightly longer than on Earth. Martian clocks are therefore designed to *time slip* at midnight for 39 minutes, 40 seconds. This allows Martians to use a 24 hour clock which they are accustomed to.

A Martian year, abbreviated MYr, contains 668.6 sols, or 689 days by a terrestrial metric. That means there are 88775.245 seconds in a Martian year with 1.8876712 Martian years for every Terran year (1 year, 320 days, and 18.2 hours).

The L_s system is used to designate the solar longitude, the angle Mars makes with respect to the Sun. This is measured from the northern hemisphere with the vernal or spring equinox ($L_s = 0^\circ$). Therefore $L_s = 90^\circ$ is the summer solstice; $L_s = 180^\circ$, the autumn equinox; and $L_s = 270^\circ$ the winter solstice.

There are no timezones. All clocks are set to those of the first settlement location.

¹ A *yestersol* being the sol preceding the current one.

8 Gameplay

9 Selected Technology

It is generally a bad engineering practise to re-invent the wheel by creating something in a vacuum when there are already many usable components. Avaneya's dependent technology is examined here, grouped by category. It is sometimes explained in their maintainer's own words.

We always try to stick to using free software whenever possible. The only time we might use a non-free tool is to use it as a model in designing a suitable replacement. Another time would be where writing a replacement would be far too difficult and we have no choice.

9.1 Audio

- We use *libmikmod* for playback of tracker music modules. Some people were requesting tracker support, and since it should not take very much effort to integrate into the engine's *AudioManager* subsystem, it seems reasonable. The library also needs to be built thread safe.
- *OpenAL* is used for 3D spatial audio rendering. Actual decoding of audio data is done through other APIs.
- The *SDL* library is used to provide audio decoding through its *SDL_audio* API. It decodes popular formats like Ogg Vorbis.

9.2 Developer Tools

- The *CppUnit* library is a unit testing framework module for the C++ programming language, used only when the engine is built in debug mode.
- The *GNU Autotools* are used to reconcile and harmonize the idiosyncrasies of different platforms. It makes portability far more straightforward than without it.
- The *GNU Compiler Collection* is used for the engine's dependency calculation, compilation, and linking, among other things. Ports of virtually all of its compilers are available everywhere.
- *GNU gettext* is used for localization and translation to different human languages by making it possible to substitute strings that are marked for translation in our C++ code with a string from another language.
- The *libebml* library is the backbone of the Matroska multimedia container format. EBML stands for *Extensible Binary Meta Language*. It is often thought of as a binary analogue to the XML format, though not a complete analogue because, unlike XML, the schema must be known in advance. Therefore, it is ideally suited to be read and written by machines and not humans. See [Section 11.1 \[AresPackages\]](#), [page 22](#) for details on how we are using it in this project.

- The *streflop* library (*STandalone REproducible FLOating-Point*) allows you to control how floating point computations are done in C++. The goal is to make programs give reliable and reproducible results. This is important because differences in machine generated code, numeric handling libraries, dedicated hardware floating point processors, optimizations, and so on, can yield results that are inconsistent across different environments.
- The *xmlstarlet* tool is used to validate XML against a schema. It is useful to check AresPackage manifests and the engine event definition for syntactical errors.

9.3 Graphics

- All modelling is generally done with *Blender*, but modellers are free to use whatever free modelling application they like, as long as it supports common free formats. One popular alternative is *Wings 3D*. Ultimately though, all models have to importable into Blender since we are dependent on the OGRE 3D exporter for it in order to integrate with the rendering engine.
- *Caelum* is a weather generation plugin for the OGRE 3D rendering engine. It is used to provide weather effects wherever necessary on Mars.
- The *CEGUI* library allows graphical user interfaces to be built on top of the *OGRE 3D* rendering engine. This is necessary for the in game graphical user interface. CEGUI user interfaces are defined through Lua scripts accessing its Lua interface.
- *Hydrax* is a fluid dynamics plugin for the OGRE 3D rendering engine. It is used to provide fluid effects wherever necessary on Mars.
- The *OGRE 3D* rendering engine is a powerful, cross-platform, API generally aimed at game developers. It has a rich and simple to use API with a plethora of plugins available for it. It is strictly a rendering engine and does not, however, cover input, audio, and other standard game engine subsystems.
- The *OpenGL* library provides the rendering backend for the OGRE 3D rendering engine. Although the latter supports other backends, it is very difficult to write and maintain shaders for all of them. OpenGL is ubiquitous these days, can do virtually everything Direct3D can, and does not hold you hostage to any specific platform.
- The *OpenGL Extension Wrangler Library* library (*GLEW*, helps in querying and loading OpenGL extensions. It provides efficient run-time mechanisms for determining which OpenGL extensions are supported on the target platform. All supported OpenGL extensions are exposed in a single header file, which is machine-generated from the official extension list.
- *Terrain* is another plugin for the OGRE 3D rendering engine allowing for, as the name suggests, terrain generation.

- The *SDL* library is used to provide image and font loading through its *SDL_image* and *SDL_pango* APIs respectively.

9.4 Input

- The *SDL_haptic* API is used for providing force feedback, provided the API is available and the user's input device supports it. SDL version 1.3 or greater is required.
- The *SDL_input* API is used to provide input handling.

9.5 Networking

- The *ENet* library provides the low level library which Avaneya's multiplayer protocol is built upon. Its purpose is to provide a relatively thin, simple and robust network communication layer on top of UDP (User Datagram Protocol). The primary feature it provides is optional reliable, in-order delivery of packets. It omits certain higher level networking features such as authentication, lobbying, server discovery, encryption (which we provide via GnuTLS), or other similar tasks that are particularly application specific so that the library remains flexible, portable, and easily embeddable.
- The *GNU Transport Layer Security Library* (GnuTLS) is a free software implementation of the SSL and TLS protocols. It is used to provide encryption at the application level to make it more difficult for an uninvited third party to be a nuisance.

9.6 Operating System

- *Apport* intercepts program crashes, collects debugging information about the crash and the operating system environment, and sends it to us in a standardized format by integrating directly into Launchpad. It is available only on supported GNU operating systems.
- The *D-Bus* (Desktop Bus) API is a simple inter-process communication (IPC) system for software applications to communicate with one another. Avaneya can use it to determine the state of the user's network connection, among other things. It is available only under supported POSIX compliant operating systems.

9.7 Physics

- *OgreBullet* is a plugin adding physics support to the OGRE 3D rendering engine. It is built upon *Bullet*, a free physics engine featuring 3D collision detection, soft body dynamics, and rigid body dynamics. It is used in games, and in visual effects in movies.

9.8 Scripting

- The *libtolua++* library is used to integrate our C++ code with Lua. It is an updated

replacement to the older toLua tool. It makes it possible for engine code to invoke Lua code, and vice versa.

- *Lua* is used to provide scripting support to the AresEngine. The runtime environment needs to have *package.loadlib* present.
- The *lua-gettext* API is a Lua package that acts as a Lua wrapper for gettext bindings.
- The *lua-xgettext* tool is a small program for message extraction of marked strings from Lua code so they can be made available for translation to other languages. It is similar to *GNU xgettext*, but far more primitive; it just extracts the strings and prints them out, without any additional information. It was written, because GNU ‘xgettext’ did not support Lua at the time.

10 Information For Everyone

10.1 Socioeconomic Modelling

Avaneya relies heavily on actual scientific research to substantiate the quantitative models it encodes.

The Equality Trust is an organization that aims to reduce income inequality through a programme of public and political education designed to achieve a widespread understanding of the harm caused by income inequality, public support for policy measures to reduce income inequality, and the political commitment to implementing such policy measures.

The research it has aggregated powerfully reasserts the view that poverty, despair, and crime are not accidents, are well predictable, and that this is obvious by their asymmetrical spatial distribution. The research suggests that there is no such thing as human nature, but only human needs, and consequently human behaviour.¹

The Equality Trust supports this conclusion by providing an invaluable repository of research that lays the foundation for the quantitative models necessary for a simulation like Avaneya.

<http://www.equalitytrust.org.uk/resources/publications>

¹ See *Zeitgeist: Moving Forward* in [Section 3.1 \[Resources For Everyone\]](#), page 4.

11 Information For Engineers

11.1 AresPackages

11.1.1 Purpose

The AresEngine uses a custom game archive format built using EBML, an extensible language akin to XML, but tailored to handling binary data. The Matroska multimedia container is its most prominent client.

You might be wondering why a game engine requires its media to be delivered to it in a custom archive format, as opposed to being exposed "naked" directly through the platform's native file system. There are a number of benefits to using a custom archive format, as well as EMBL specifically.

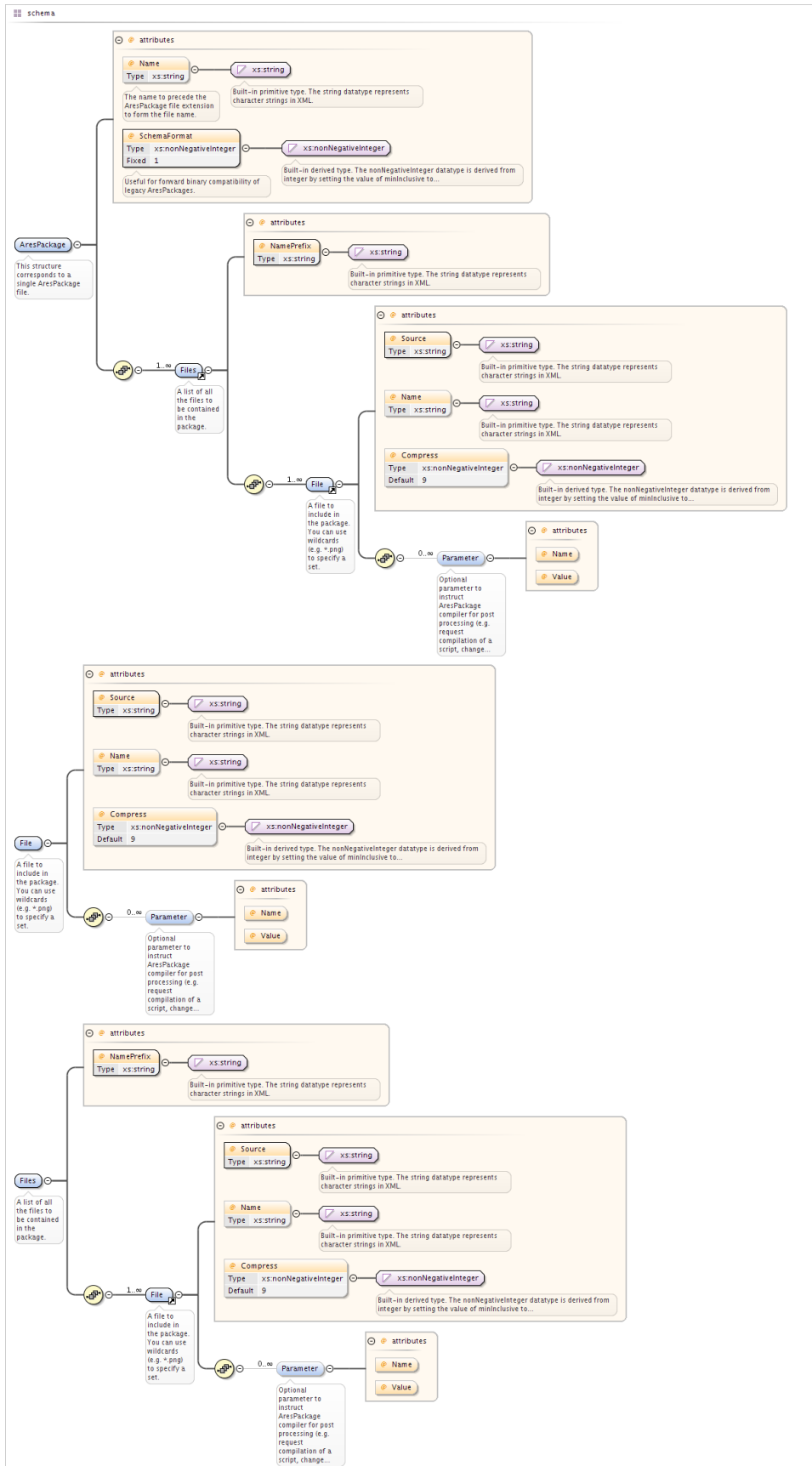
- The EMBL format enables extendability for future format changes.
- It can be easier to distribute a few files containing many, than many to the end user.
- As a single file (*.AresPackage), file seek, open, and load times are reduced.
- Providing a layer of abstraction between the actual raw data and the client that requires it allows us to decompress compressed data on the fly. This results in a smaller file, which means a faster disk to RAM transfer. Remember that the disk is slow, while the CPU is much faster.

11.1.2 Usage

The *ares-package* tool takes a package manifest file (XML) describing the contents of the package to output. The tool archives the requested files, along with whatever settings that may be required, and outputs the package.

11.1.3 Design

A package manifest is checked against an XML schema (AresPackage.xsd) to verify it is syntactically correct. AresSamplePackage.xml is a sample package. The following diagram is a graphical representation of the schema.



11.2 Coding Standards

Try to abide by the project coding standards whenever possible. It is easier to adapt to a new coding standard in a project when that standard is uniformly applied to it, as opposed to where everyone applies their own. Try to apply the following conventions whenever working in C++, as well as whatever is practical to carry over to shaders, scripts, and elsewhere.

11.2.1 Legal

Headers (*.h), implementations (*.cpp), and other code, as defined under [Chapter 13 \[Licensing Rationale\]](#), [page 59](#), should have prefixed the following legal notice. Adapt the syntax for comments as necessary for the given machine environment (e.g. shader, makefile, etc.).

```
/*
    AresEngine, a 3D game engine.
    Copyright (C) 2011 Kshatra Corp <kip@thevertigo.com>.

    Public discussion on IRC available at #avaneya (irc.freenode.net)
    or on the mailing list <avaneya@lists.avaneya.com>.

    This program is free software: you can redistribute it and/or modify
    it under the terms of the GNU General Public License as published by
    the Free Software Foundation, either version 3 of the License, or
    (at your option) any later version.

    This program is distributed in the hope that it will be useful,
    but WITHOUT ANY WARRANTY; without even the implied warranty of
    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
    GNU General Public License for more details.

    You should have received a copy of the GNU General Public License
    along with this program. If not, see <http://www.gnu.org/licenses/>.
*/
```

11.2.2 Mathematical

- All angles should be worked with in degrees, not radians, whenever possible.
- All floating point calculations should be done in single precision whenever possible.

11.2.3 Formatting

- English (Canadian) should be used for everything non-localized and saved in UTF-8 whenever possible.
- Always use spaces and not tabs. You can use the latter, if you like, but just as long as your editor converts them to spaces when saving.
- Each level of nesting should be indented by four spaces. The exception to this is when a prefix operator precedes an identifier, in which case it should be indented so that the identifier begins at a four space interval.

Right:

```
float SomeFunction()
{
    int i = 0;
    ++i;
    return 1.0f;
}
```

Wrong:

```
float SomeFunction()
{
    int i = 0;
    ++i;
    return 1.0f;
}
```

- You do not need to indent to accommodate a namespace when an entire source file is enclosed within one. If only part of the source is enclosed in a namespace, then those parts affected should be indented.
- An opening brace appears on the next line as preceding code.

Right:

```
float SomeFunction()
{
    for(int x = 0; x < 100; ++x)
    {
        ...
    }
}
```

Wrong:

```
float SomeFunction() {
    for(int x = 0; x < 100; ++x) {
        ...
    }
}
```

- Put spaces around binary operands

Right:

```
A = X + Y;
if(X == Y)
    ...
```

Wrong:

```
A = X+Y;
if(X==Y)
    ...
```

- There is no space between unary operators and the affected variable's name.
- When accessing an array, there is no space between the array name and the opening left bracket.

Right:

```
int n[100];
y = n[3];
```

Wrong:

```
int n [100];
y = n [3];
```

- The if, for, and while keywords are not followed by a space separating them and the left parenthesis.

Right:

```
if(x == 4)
    return;
```

Wrong:

```
if (x == 4)
    return;
```

- When declaring variables, align the variable names on the same column evenly divisible by four.

Right:

```
int      Count;
bool     Complete;
Window   MainWindow;
```

Wrong:

```
int Count;
bool Complete;
Window MainWindow;
```

- When calling a function, definitions and function declarations should have no space between the function name and the following left parenthesis.

Right:

```
int Add(int a, int b)
{
    ...
}
```

```
int i = Add(4, 5);
```


Wrong:

```
int Add (int a, int b)
{
    ...
}
```

```
int i = Add (4, 5);
```

- When calling a function or making a function declaration, no space appears after the left parenthesis or before the right parenthesis.

Right:

```
foo(x, y);
bar(z);
baz();
```

Wrong:

```
foo( x, y );
bar( z );
baz( );
```

- A brace preceding or following an else keyword appears on the same line as the else. A statement following an else keyword appears on the same line as the else.

Right:

```
if(x == 4)
{
    ...
}
else
    ++y;

if(x == 4)
{
    ...
}
else
{
    ...
}
```

Wrong:

```
if(x == 4) {
    ...
}
else ++y;

if(x == 4) {
    ...
} else ++y;

if(x == 4) {
```

```

    ...
}
else {
    ...
}

```

- A brace preceding a catch keyword appears on a separate line as the catch.

Right:

```

try
{
    ...
}

catch(Error SomeError)
{
    ...
}

```

- A value in a return statement is parenthesized where it contains more than one term.

Right:

```

return x;
return (a + b);

```

Wrong:

```

return (x);
return a + b;

```

- If the body of an if, for, while or similar statement consists of a single statement, the statement does not need to be surrounded by braces.

Right:

```

if(x == 3)
    ++x;

```

11.2.4 Naming

- Do not use Hungarian notation. We prefix objects to denote scope only.

Member of a global namespace:

```
g_Wheels
```

Member of a structure or class:

```
m_Wheels
```

Static member of a structure or class:

```
ms_Wheels
```

- Class and object names should be intuitive, try to avoid abbreviations, and each word should begin with a capital letter. Modern storage mediums can afford to spare brevity, allowing for greater clarity.

Right:

```
// Class for abstracting a camera interface...
class Camera
{
    ...
};

// Create a camera...
Camera LogitechCamera;
```

Wrong:

```
class cam
{
    ...
};

cam log;
```

11.2.5 Structure

- Source lines may be up to 100 characters long. (You can configure gedit to display a margin at 100 characters; that may help you follow this convention.)
- Functions or methods should be broken down into other functions or methods if they get too long and this is reasonable to do.
- Use *assert()* to check your assumptions for things that ought to always be true. Do not abuse it for situations where it is reasonable for a condition to not be true, such as a socket connection failure or a file that could not be opened.
- Follow the *GNU Coding Standards* as much as reasonably possible, save the code formatting points made in this handbook. There is a great deal of wisdom in it.
- A comment which indicates task which needs to be done at some point should look like this:

```
// TODO: Check portability here...
```

11.2.6 Comments

- All comments should be written in Canadian English since nearly all programmers in all countries can read that. If you cannot do that, write them as best you can and have someone help you rewrite them.
- Comments should begin with a single space, then a capital letter and end with a trailing ellipses.

Right:

```
// Load the image...
GrayImage = cvLoadImage(Path.mb_str(), CV_LOAD_IMAGE_GRAYSCALE);
```

Wrong:

```
GrayImage = cvLoadImage(Path.mb_str(), CV_LOAD_IMAGE_GRAYSCALE); //load the image.
```

- The farther left the comment, the higher level and abstract what you are trying to do is. The farther right, the more detailed they are. You can think of a given level of indentation as elaborating on how to carry out what was described at a higher (less indented) level. It should be possible in many cases to strip away all the code, except the comments, and still understand what it is that you were trying to do.

Before:

```
// Analyze single image...
void AnalysisThread::AnalyzeImage(wxString Path)
{
    // Variables...
    IplImage *GrayImage = NULL;
    wxString TempString;

    // Reset the tracker, if not already...
    Frame.Tracker.Reset(0);

    // Load the image...
    GrayImage = cvLoadImage(Path.mb_str(), CV_LOAD_IMAGE_GRAYSCALE);

    // Failed to load media...
    if(!GrayImage)
    {
        // Alert...
        wxLogError(wxT("Unable to load image."));

        // Abort...
        return;
    }

    // Feed into tracker...
    Frame.Tracker.Advance(pGrayImage);

    // Cleanup gray image...
    cvReleaseImage(&GrayImage);
}
```

Stripped:

```
// Analyze single image...

// Variables...

// Reset the tracker, if not already...

// Load the image...

// Failed to load media...
```

```

// Alert...

// Abort...

// Feed into tracker...

// Cleanup gray image...

```

- A single blank line should appear between each pair of functions or methods.
- Do not use a blank line after an opening brace or before a closing brace.
- Do not use duplicate blank lines.

11.2.7 Developer Tools

You are welcome to use any editor or IDE you like, provided it does not require non-standard IDE-specific project files. See [Section 9.2 \[Dependent Technology Developer Tools\]](#), page 17 for a list of all of the developer tools that are required for the project.

11.3 Architecture

The *AresEngine* is responsible for delivering Avaneya. It is designed by the Avaneya crew to provide for the game, but kept architecturally general enough to lend itself to other projects of similar technical requirements (e.g. a city builder / real time strategy genre). If you do not know what a game engine is, Jason Gregory's book *Game Engine Programming*,¹ is highly recommended.

The AresEngine is divided up into many subsystems, each responsible for a given logical task. These include audio, responding to input devices, updating the graphical user interface, artificial intelligence, and so on.

Umbrello was used to come up with the architectural design. At this time, you may need to use the latest Umbrello built from source because the task at hand was so demanding, only Umbrello's bleeding edge was able to handle it. Unfortunately the pre-compiled binary available in most distributions had many show stopper bugs in it that would have made work impossible. Even building from source, it still has many problems. Nevertheless, it was the most mature and functional free design tool available at the time and may still be. We are very grateful to the Umbrello crew who were especially helpful in fixing bugs as they were discovered through the engine design stress testing it.

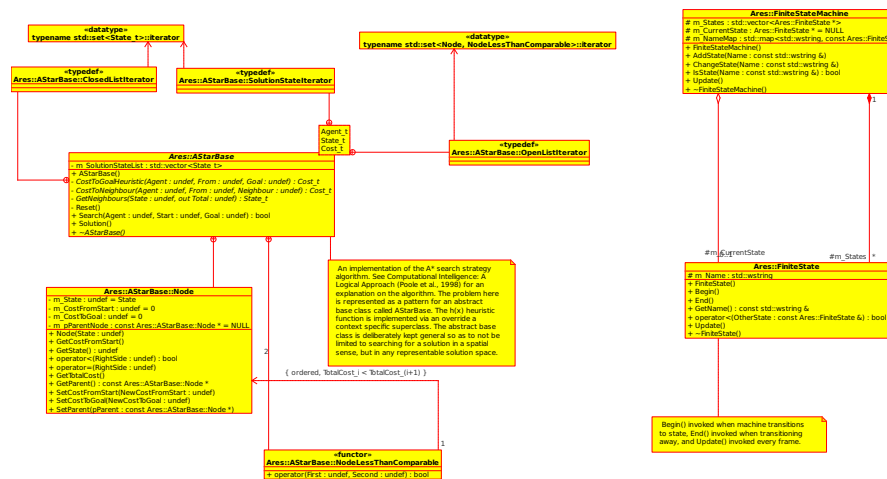
The following architectural diagrams were pulled from the AresEngine's Umbrello project file automatically at the time this handbook was compiled on 16 May 2011.

¹ See [\[Resources for Engineers\]](#), page [\[undefined\]](#) for details.

11.3.1 Artificial Intelligence

This diagram concerns itself with useful artificial intelligence algorithms, though not of an entire subsystem itself.

It contains an implementation of the A* search strategy algorithm.¹ The problem here is represented as a pattern for an abstract base class called *AStarBase*. The $h(x)$ heuristic function is implemented via an override a context specific superclass. The abstract base class is deliberately kept general so as to not be limited to searching for a solution in a spatial sense, but in any representable solution space (e.g. time, language, etc.).



The Artificial Intelligence subsystem.

Figure 11.1

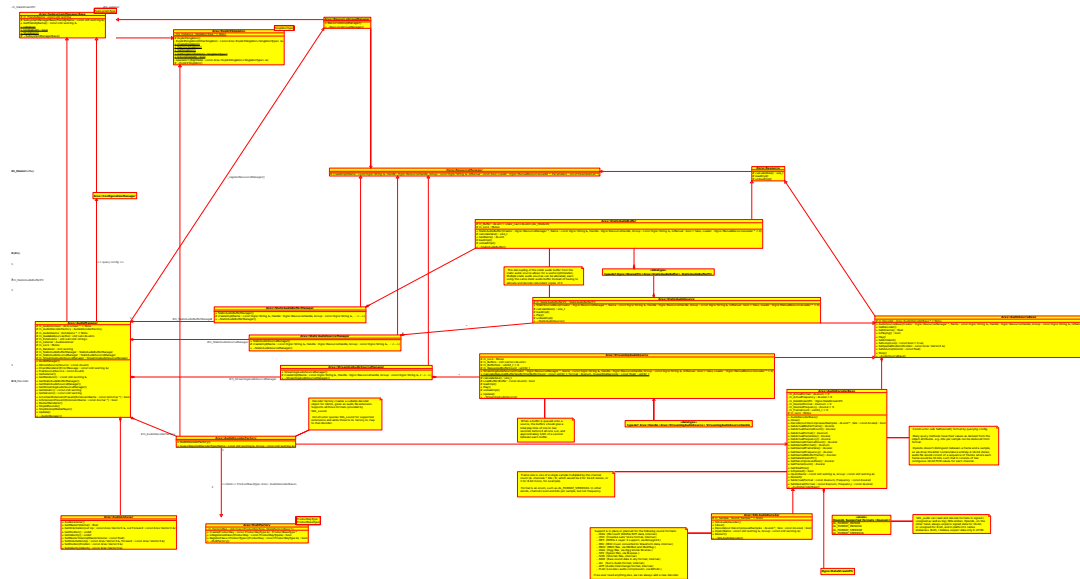
¹ See *Computational Intelligence: A Logical Approach*, Poole et al., 1998 for an explanation on the algorithm.

11.3.2 Audio

The *AudioManager* subsystem is responsible for all tasks related to audio. Audio playback is divided into streaming and static audio sources. Streaming sources need to be continuously updated by loading new data from disk, decompressing it, and playing it. A common example would be music or narrative. Static sources are usually smaller and only need to be played once before being freed. Static source examples would be things like the sound of an object contacting another.

Decoding is done through an appropriate subclass of an *Ares::AudioDecoderBase* abstract class, instantiated via the *Ares::AudioDecoderFactory* class. Most decoding is done through a subclass of the aforementioned via *SDL_audio*.

Playback is accomplished through OpenAL.



The audio subsystem.

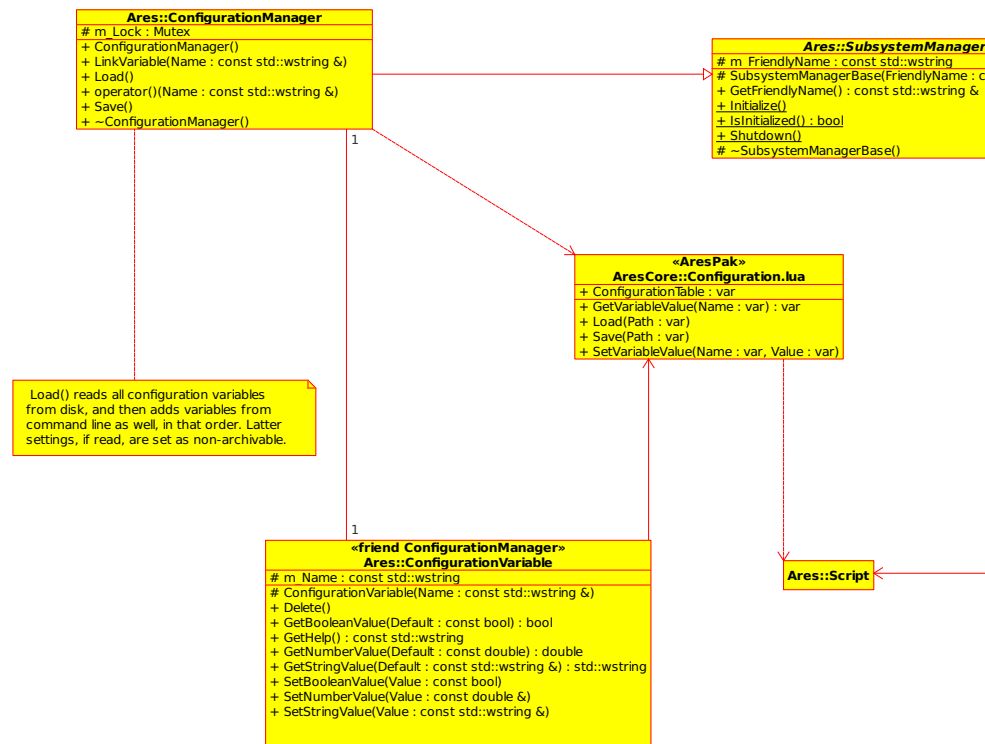
Figure 11.2

11.3.3 Configuration

The *ConfigurationManager* subsystem is responsible for storing all user configuration. Its *Load()* method reads all configuration variables from disk, and then adds variables from command line as well, in that order.

Latter configuration variables read are set as non-archivable. A non-archivable configuration variable is one that is not saved to disk.

Much of the manager is implemented in Lua because of its excellent database handling.



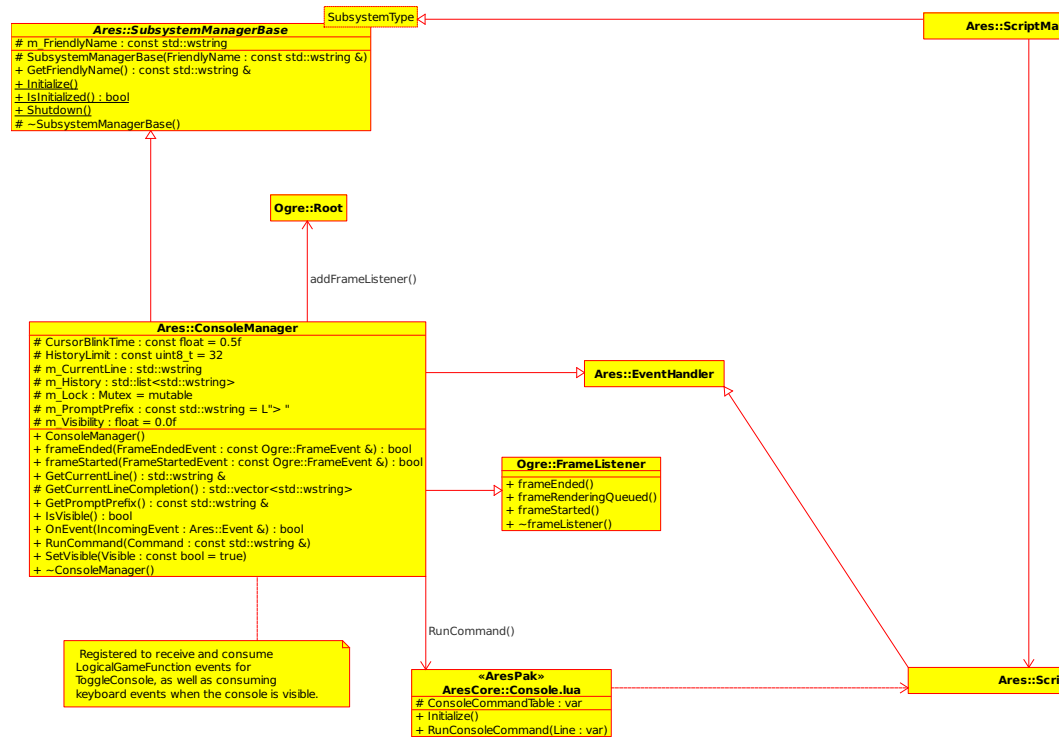
The configuration subsystem.

Figure 11.3

11.3.4 Console

The in game console allows users to access aspects of the game engine at runtime. This is useful for debugging or other purposes.

Console commands are implemented in Lua.



The console subsystem.

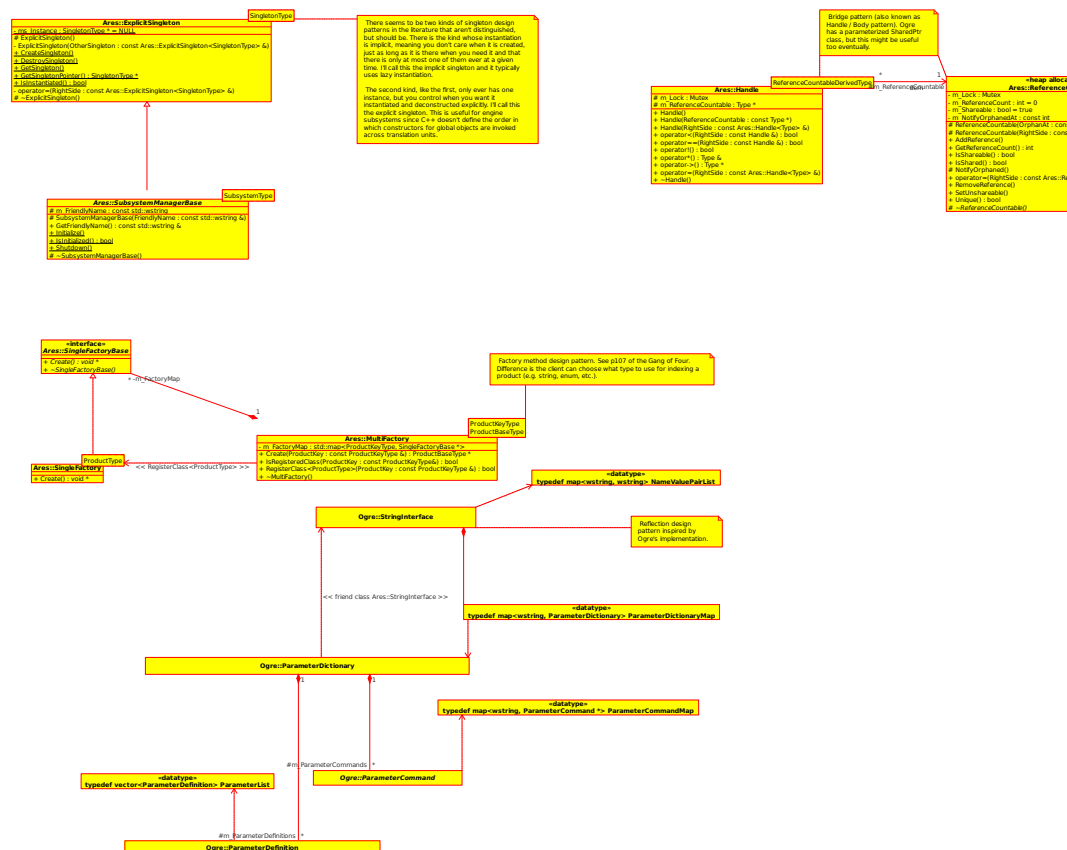
Figure 11.4

11.3.5 Design Patterns

This diagram contains a number of common place design patterns found in many software projects, such as the singleton and multi-factory.

There seems to be two kinds of singleton design patterns in the literature that are not distinguished, but should be. There is the kind whose instantiation is implicit, meaning you don't care when it is created, just as long as it is there when you need it and that there is only at most one of them ever at a given time. We refer to this as the *implicit singleton* and it typically uses lazy instantiation.

The second kind, like the first, only ever has one instance, but you control when you want it instantiated and deconstructed explicitly. We refer to this as the *explicit singleton*. This is useful for engine subsystems since C++ does not define the order in which constructors for global objects are invoked across translation units. This is important to consider, given that the order of subsystem initialization is very important (e.g. resource management must precede audio management).

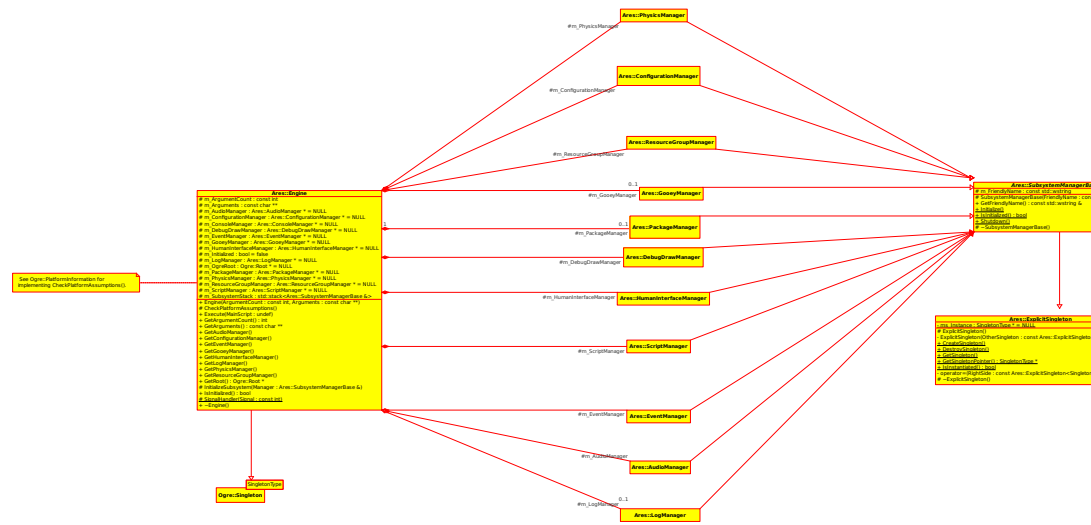


Useful design patterns.

Figure 11.5

11.3.6 Engine

This diagram captures the engine at the highest level and provides an overview of all components in the most abstract sense.



The engine overview.

Figure 11.6

11.3.7 Events

The *EventManager* is responsible for intercommunication between various engine components and scripts. The steps for working with events are as follows.

An event handler can either contain an *Ares::EventHandler* class, or can derive from it and override the *OnEvent()* method. It registers interest in one or more events via the *Register()* method:

```
EventHandler::Register("some_event");
```

Next, to create and enqueue an event, one would do as follows:

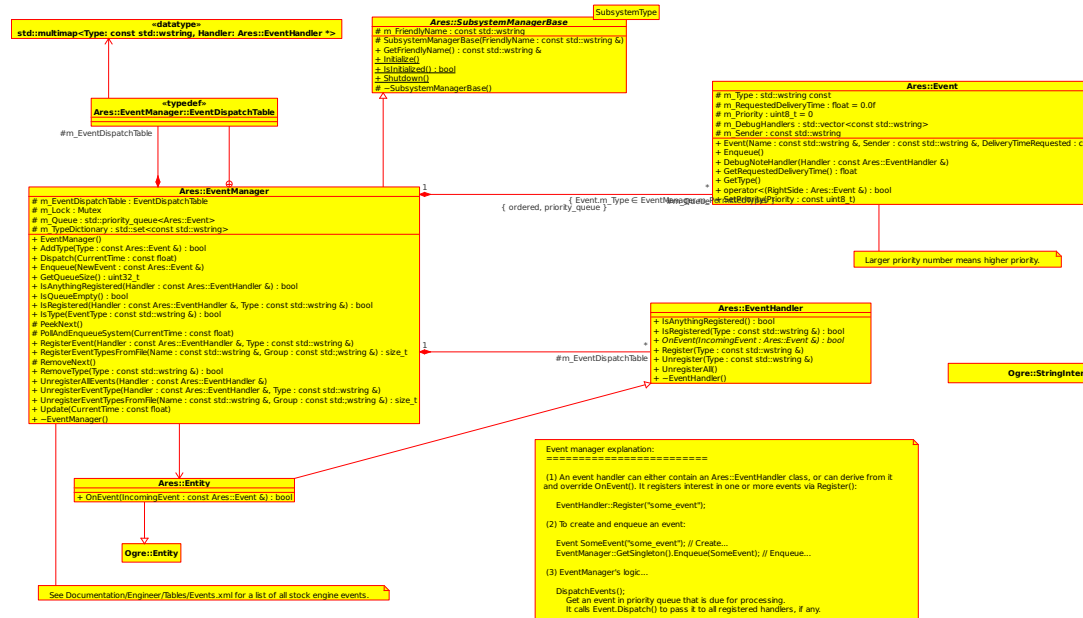
```
// Create a some_event event...
Ares::Event SomeEvent("some_event");

// Enqueue event...
Ares::EventManager::GetSingleton().Enqueue(SomeEvent);
```

Every frame, the *EventManager::DispatchEvents()* method is called to pump the message queue. It will get an event from the internal priority queue that is due for processing. Once an event is fetched, it calls *Event.Dispatch()* to pass it to all registered handlers so that they may do whatever it is that they would like to do when that event occurs.

Events.xsd defines the XML schema that the engine uses at runtime to check builtin event types (Events.xml) against. This is done to ensure the event types defined are syntactically correct and self documenting.

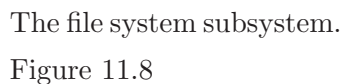
Events.xml contains a list of built-in event types, game engine mod agnostic, for fundamental event types. These include things like input device and window manager events.



The event subsystem.

Figure 11.7

Accessing files on disk for textures, shaders, scripts, models, sounds, and so on, requires an intermediary in most game engines. The AresEngine is no different. Files are stored inside of an EBML encoded container format called an AresPackage. The EBML format was selected because it enables extendability for future format changes and allows arbitrary metadata. It also has performance advantages because only a single file (*.AresPackage) needs to be located for a group of files contained within. This reduces file seek and access times. Also, because the AresPackage files can be compressed, it means that they take less time to access their payload because transferring a small file from disk and decompressing in RAM is faster than transferring a large uncompressed file from disk to RAM.



11.3.9 Gooney

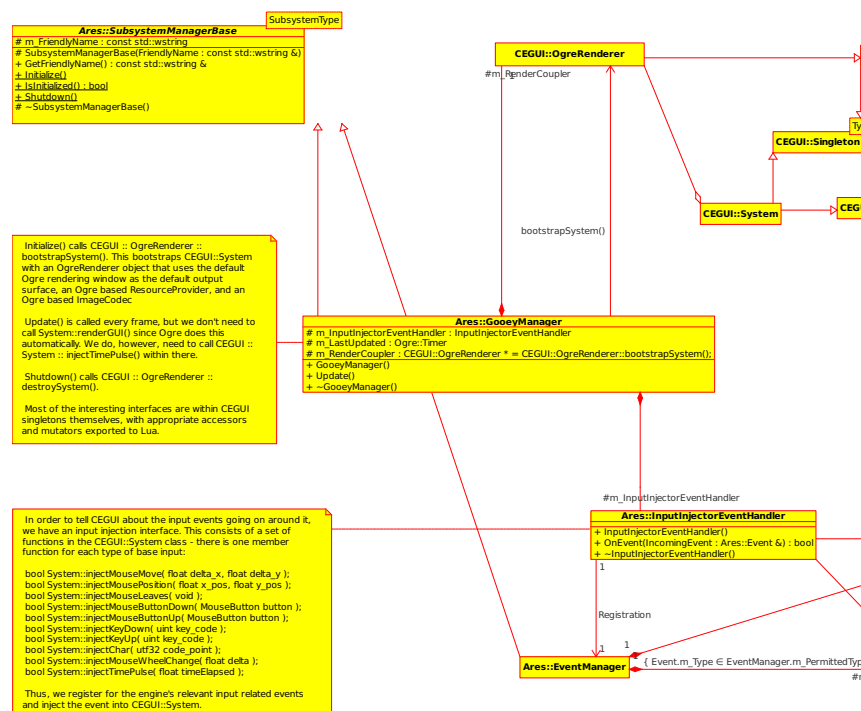
The *GooneyManager* is responsible for managing all graphical user interface overlays and user interaction with them. Its underlying functionality is provided by CEGUI which integrates well into OGRE 3D.

GooneyManager::Initialize() calls *CEGUI::OgreRenderer::bootstrapSystem()*. This bootstraps *CEGUI::System* with an *OgreRenderer* object that uses the default Ogre rendering window as the default output surface, an Ogre based *ResourceProvider*, and an Ogre based *ImageCodec*.

Update() is called every frame, but we do not need to call *System::renderGUI()* since Ogre does this automatically. We do, however, need to call *CEGUI::System::injectTimePulse()* within there.

Shutdown() calls *CEGUI::OgreRenderer::destroySystem()*.

Most of the interesting interfaces are within CEGUI singletons themselves, with appropriate accessors and mutators exported to Lua.

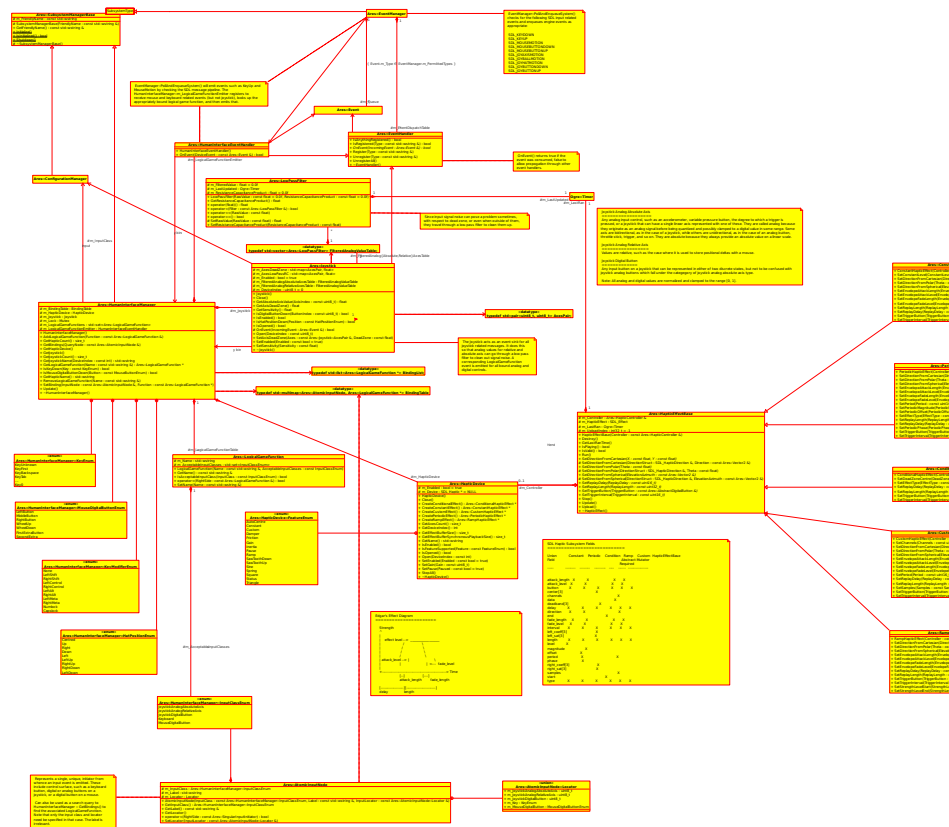


The graphical user interface subsystem.

Figure 11.9

11.3.10 Human Interface

The *HumanInterfaceManager* takes care of routing all events from input devices to the appropriate code that has registered to be notified of that event. In that sense, the manager is mostly unidirectional, but also allows to send information the other way back to the input device if it supports a haptic interface (sometimes called force feedback).



The human interface subsystem.

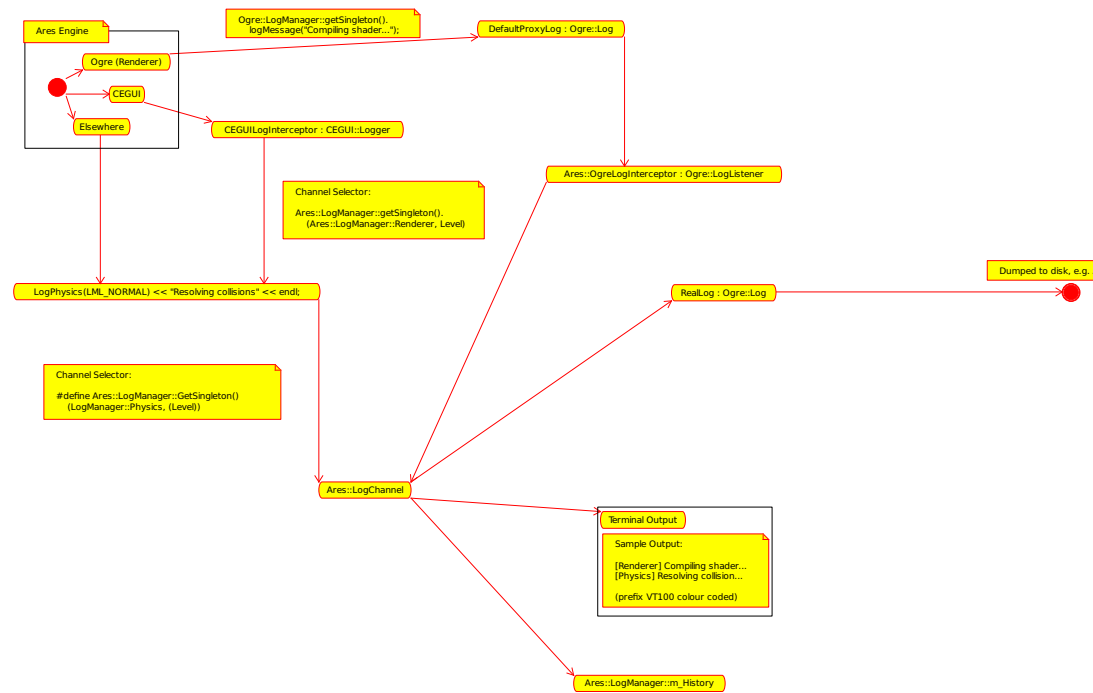
Figure 11.10

The *LogManager* is responsible for providing a central logging subsystem for all the other subsystems.



Figure 11.11

Since message can be emitted from any number of places, such as within the engine, or within some other externally linked component, such as CEGUI or OGRE, messages have to be intercepted and channeled through the *LogManager*. The following diagram makes this more clear.

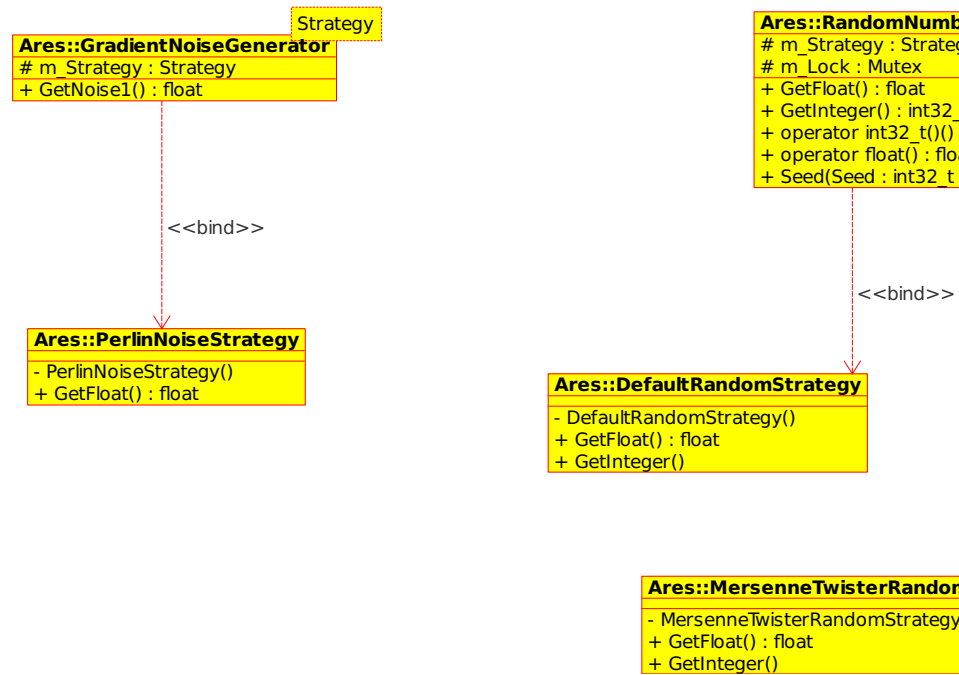


The logging pipeline.

Figure 11.12

11.3.12 Mathematical

Most mathematical routines for matrices and other linear algebra are provided by OGRE 3D. Some things like more advanced random number generation have to be implemented ourselves here.

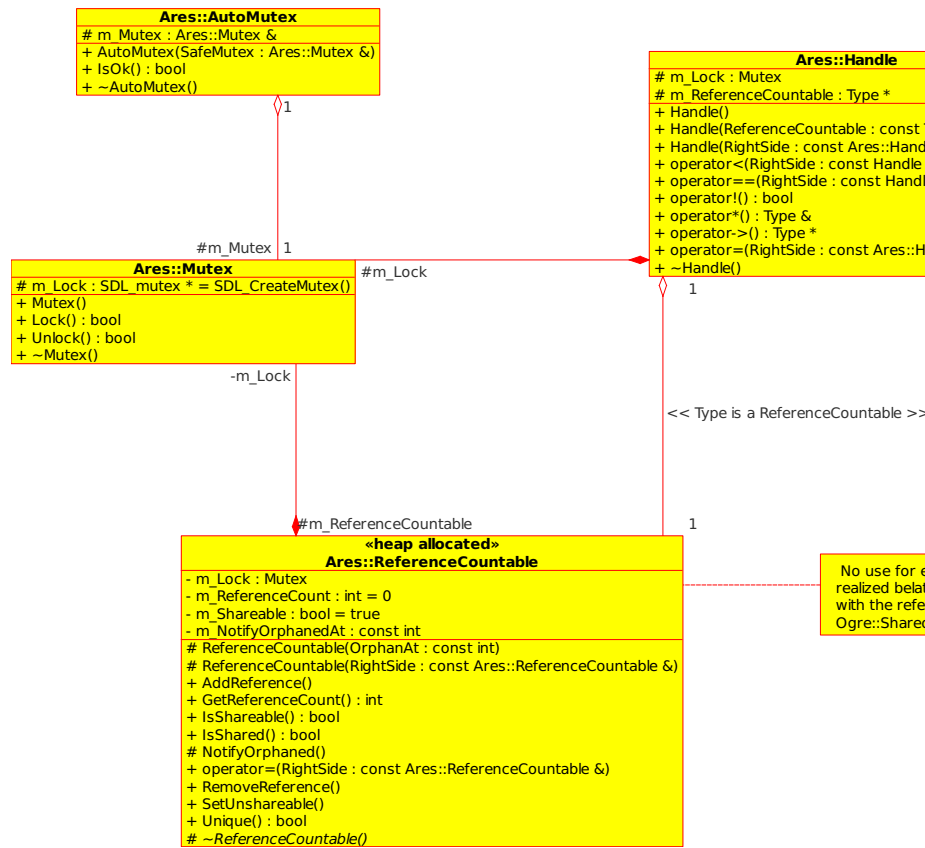


Various mathematical facilities.

Figure 11.13

11.3.13 Miscellaneous

This is a catch-all for classes and functions that had no business elsewhere.

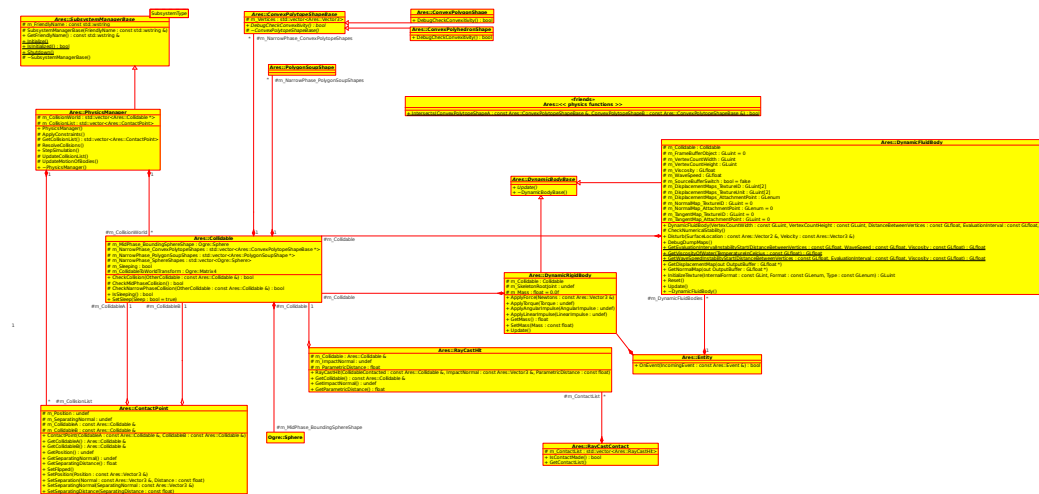


Miscellaneous engine components.

Figure 11.14

11.3.14 Physics

The *PhysicsManager* needs to be seriously refactored since *OgreBullet* was found as a viable option for physics support. This is the original, incomplete, design before that decision was made.



The scripting subsystem.

Figure 11.15

The *ResourceManager* is responsible for ensuring that whenever code needs a piece of game data, such as a model or animation, all of its requisites are loaded, in the correct order, only once, and take up only as much room as they absolutely must. The *ResourceManager* and the *FileManager* communicate to help the former locate what it needs physically on disk.

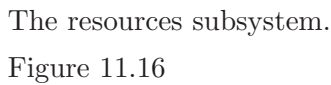
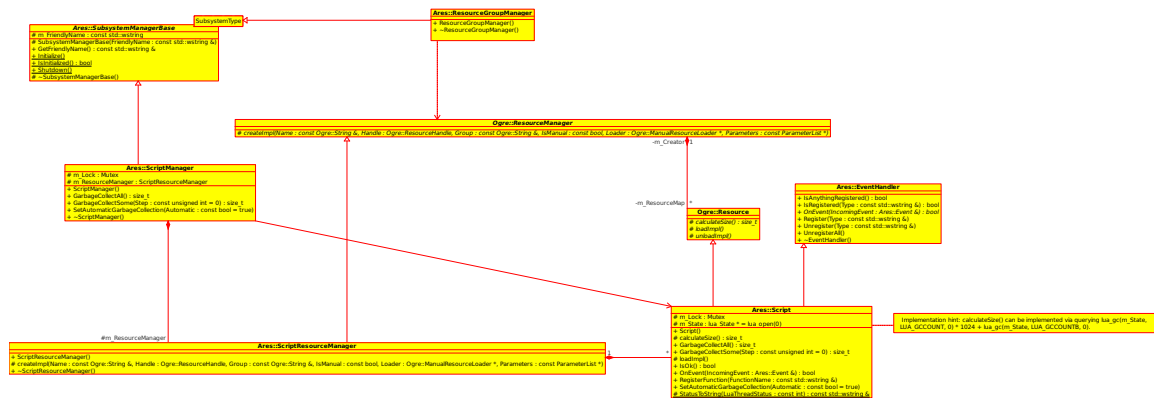


Figure 11.16

11.3.16 Scripting

The *ScriptManager* is responsible for exposing all useful aspects of the game engine to the game scripters. Since the engine itself is defined in code, the role of defining what makes Avaneya, Avaneya, is left to script writers.



The scripting subsystem.

Figure 11.17

12 The Project Crew

12.1 If You Are New

If you are new to the Avaneya crew, welcome! You will probably enjoy working on the project and the sense of community *software libre* projects tend to create. To get you started, here are a couple things you should probably do as soon as you can.

1. Read this handbook, which you are already doing.
2. Get on the low volume announcement mailing list. See [Section 12.5.1 \[Mailing Lists\]](#), [page 54](#) for details.
3. Join the Avaneya Crew on Launchpad. Membership requires approval. If you already received an invite, great. Otherwise, feel free to ask us. There is lots of room in the project for more talent. See [Section 12.5.3.1 \[Launchpad Crew\]](#), [page 56](#) for details.
4. Also join the private discussion mailing list at the aforementioned link. Once on, I recommend you email the list and let everyone know a bit about yourself. They are all good people and will be happy to see a new face. You will be joining fellow musicians, engineers, and artists. See [Section 12.5.1 \[Mailing Lists\]](#), [page 54](#) for details.

12.2 Specialities

Avaneya is a fairly large, creative, and collaborative project, calling upon a variety of different disciplines. These are the different capacities Avaneya contributors fall into. There is no reason why someone who wishes to work in more than one capacity cannot. Many do so already.

2D Artist 2D artists work on the graphical user interface managed by cegui, texturing for models, and other things of that sort. Their areas of expertise range from conceptual art, storyboarding, font design, bump maps, and procedural textures (materials). They can work with a range of software from FontForge to GIMP.

Audio Engineer

These folks creatively can take unassuming sounds and transform them into things usable for science fiction game. An example would be sampling the squeak of a chair or a car driving by with a high end portable recorder and remixing it into the sound of a nuclear electric ion-drive propulsion system. You might find them working with software like Ardour, Rosegarden, and portable high resolution audio recorders.

Cinematic Artist

Cinematics play an important role in games. They prepare the user and set the stage in ways that would be difficult to do during normal game play. They work with tools like Blender, Cinelerra, and Lombard.

- Engineer* The engineers design, discuss, and implement the engine specification, along with writing the Lua script that drive the engine. They work mostly in the languages of C++ and GLSL. The GNU Autotools is useful in the way construction scaffolding is, and they depend on it to ensure the software stays as versatile as possible. There may be some opportunities for assembly level optimizations, but they largely work at a higher level of abstraction that the OGRE 3D rendering engine expects. They coordinate their work with the rest of the team via Bazaar and take care of distribution of pre-compiled binaries via packaging (e.g. debs).
- Modeller* Modellers produce the 3D game models the user sees during game play. They also work with the 2D artists to ensure models are properly textured. They work with Blender, Wings 3D, or any other modelling program that supports standard patent free model formats.
- Musician* The musicians create either new or provide existing tracks for the game. The music falls into two categories. The first is in game ambient music that the user passively listens to. The second is music that is more actively listened to during navigation menus, cinematics, and possibly the separately to be released game soundtrack.
- Researcher* These people provide the background information and attention to detail that makes the game rich. They have an interest in *areology* (the study of Mars), terraforming, simulation and complex modelling, social and political issues (e.g. the *Genuine Progress Indicator*), and whatever else that might be useful.
- Scripter* Scripters write code in Lua that drives and breathes life into the game engine. They will probably work with the engineers to ensure the functionality they require of the AresEngine is exposed safely.
- System Administrator* System administrators run and administer the user forum, moderate the IRC chatroom, monitor the bug tracker on Launchpad, and so on.
- Translator* Translators are what makes Avaneya available to people of different languages. They ensure cinematic subtitles and the game's GUI, and website, are properly internationalized. They work with any tools that support standard GNU gettext and language catalogues.
- Voice Actor* Cinematics and in game audio often requires real people to play a role.

Web Developer

Web developers are familiar with standards and work with things like CSS, XHTML, php, MySQL, and so on. They probably will end up coordinating with the system administrators.

Writer

Writers work closely with the researchers and other artists to provide dialogue and scripts for storyboarding and other game media.

12.3 Avaneya Code of Conduct

Version 1.0

Copyright © 2011 Kshatra Corp.

This Code of Conduct covers our behaviour as members of the Avaneya Community, in any forum, mailing list, wiki, web site, IRC channel, install-fest, public meeting or private correspondence. Avaneya governance bodies are ultimately accountable to the project lead and will arbitrate in any dispute over the conduct of a member of the community.

- **Be transparent, but don't be reckless.** This is a free software project and so we must work as transparently as possible. Having said that, no one likes to have the ending of a good film revealed to them before they have had a chance to see it for themselves. Don't reveal any elements of the storyline or game until it has been discussed and you have been given permission to do so. The crew mailing list is not open to the public and is reserved for Avaneya crew only. Topics range from game storyline spoilers, unreleased conceptual art, security vulnerabilities, and so on. What goes on in this list stays in this list.
- **Be considerate.** Our work will be used by other people, and we in turn will depend on the work of others. Any decision we take will affect users and colleagues, and we should take those consequences into account when making decisions. Avaneya may one day have millions of users and thousands of contributors. Even if it's not obvious at the time, our contributions to Avaneya will impact the work of others. For example, changes to code, infrastructure, policy, documentation, and translations during a release may negatively impact others' work.
- **Be respectful.** The Avaneya community and its members treat one another with respect. Everyone can make a valuable contribution to Avaneya. We may not always agree, but disagreement is no excuse for poor behaviour and poor manners. We might all experience some frustration now and then, but we cannot allow that frustration to turn into a personal attack. It's important to remember that a community where people feel uncomfortable or threatened is not a productive one. We expect members of the Avaneya community to be respectful when dealing with other contributors as well as with people outside the Avaneya project and with users of Avaneya.
- **Be collaborative.** Collaboration is central to Avaneya and to the larger free software community. This collaboration involves individuals working with others in teams within Avaneya, teams working with each other within Avaneya, and individuals and teams within Avaneya working with other projects outside. This collaboration reduces redundancy, and improves the quality of our work. Internally and externally, we should always be open to collaboration. Wherever possible, we should work closely with dependent upstream projects and others in the free software community to coordinate our technical, advocacy, documentation, and other work. Our work should be done as transparently as possible, without ruining the elements of storytelling, and we should involve as many interested parties as early as possible. If we decide to take a different

approach than others, we will let them know early, document our work and inform others regularly of our progress.

- **When we disagree, we consult others.** Disagreements, both social and technical, happen all the time and the Avaneya community is no exception. It is important that we resolve disagreements and differing views constructively and with the help of the community and community processes when necessary.
- **When we are unsure, we ask for help.** Nobody knows everything, and nobody is expected to be perfect in the Avaneya community. Asking questions avoids many problems down the road, and so questions are encouraged. Those who are asked questions should be responsive and helpful. However, when asking a question, care must be taken to do so in an appropriate forum.
- **Step down considerably.** Members of every project come and go and Avaneya is no different. When somebody leaves or disengages from the project, in whole or in part, we ask that they do so in a way that minimises disruption to the project. This means they should tell people they are leaving and take the proper steps to ensure that others can pick up where they left off.

The Avaneya Code of Conduct is an adaptation of the Ubuntu Project's Code of Conduct, also licensed under the [Creative Commons Attribution-Share Alike 3.0](#) licence. You may re-use it for your own project, and modify it as you wish, just please allow others to use your modifications. Remember to give credit to the Ubuntu Project.

12.4 Current Members

12.5 Communication and Coordination

The team uses two primary means of intercommunicating. The first is through the crew mailing list, and the second is over IRC. The Code of Conduct applies to both. See [Section 12.4 \[Code of Conduct\]](#), page 54.

12.5.1 Mailing Lists

Mailing lists have been around for literally decades. They lack the cosmetics and ease of use of a forum, but until our website is ready, it will suffice for the time being.

Avaneya has three mailing lists. The first is a unidirectional announcement mailing list. It is low volume and only intended to communicate from the project to the general public about major project press releases. Anyone can subscribe to it.

Subscribe:

<https://www.avaneya.com/lists/?p=subscribe&id=1>

Browse the archives:

<https://www.avaneya.com/news/announcements/>

The second avaneya@lists.avaneya.com is for anyone to discuss anything related to the project. It is open to everyone.

Subscribe:

<http://lists.avaneya.com/listinfo.cgi/avaneya-avaneya.com>

Browse the archives:

<http://lists.avaneya.com/pipermail/avaneya-avaneya.com/>

The third avaneya-private@lists.avaneya.com is open only to project crew. Topics range from game storyline spoilers, unreleased conceptual art, security vulnerabilities, and so on. If you are a volunteer, you should have been sent a subscription invitation.

Subscribe:

<http://lists.avaneya.com/listinfo.cgi/avaneya-private-avaneya.com>

Browse the archives:

<http://lists.avaneya.com/private.cgi/avaneya-private-avaneya.com/>

When using either of the latter two mailing lists, you may find the following tips useful.

- Do not ever send HTML email. Some peoples' mail clients cannot render it. And even if they can, there is no guarantee it will come out the same. People with visual disabilities may not have their speech synthesizers work properly, since HTML email is much harder to parse. Others may have to pay for additional bandwidth to retrieve your email, since HTML email is larger in size than plain text.¹
- When you reply to a post, remember to reply to the list and not just the original sender privately. Unless you had intended to, the mailing list is setup so that everyone who subscribes to it may benefit from productive communication on it. Sometimes this may not happen until years later when a new subscriber searches through old archives to find a solution to a problem they were having that was solved long ago.
- When replying to a post, if you have your subscription configured to use batch digest mode,² you do not need to copy the whole digest. Just quote the minimum needed for context.
- Check the subject heading of your reply to a message posted on the list to make sure it still reflects the original post. Some mail readers, if you have batch digest mode enabled in your subscription, will change the heading to reflect the batch digest's subject heading, instead of the specific message within it you are replying to.

¹ For more information, consider reading <http://www1.american.edu/cas/econ/htmlmail.htm>

² When batch digest mode is enabled, the server will "batch" together emails into groups and then send it to you as a single compilation to cut down on the amount of email you receive.

- When you reply, remember to reply at the bottom and not at the top of the message. Top posting is generally not encouraged because it makes preservation of chronological order difficult to follow for readers.³

12.5.2 Internet Relay Chat (IRC)

IRC is among the oldest forms of realtime chat over the internet. Avaneya has a channel (**#avaneya**) on the Freenode server (**irc.freenode.net**). You can use whatever client you like, but it is recommended you use one that supports SSL.

Make sure you register your chosen nick name with the *nickserv* on Freenode. This ensures you are consistently identifiable to others in the chat room.

Whenever you would like to send someone a message publicly in the channel, you should precede your message with their nick name. This is because many people have their IRC clients configured to alert them audibly when that happens, as opposed to every time anyone says anything in the channel. Usually you only need to type the first few letters of their nick name and hit tab to have your client complete it.

12.5.3 Launchpad

Launchpad is a project started by Canonical, the company behind the Ubuntu project. It is an online service that allows for coordinated software project collaboration. It provides services for source control management, blueprints, support, bug tracking, and more. The Avaneya project, the people behind it, and the source code as it develops can be found there.

If you are a contributor, make sure you remember to add yourself to the Launchpad project crew.

Launchpad project website:

<https://launchpad.net/avaneya>

Launchpad crew website:

<https://launchpad.net/~avaneya>

12.5.3.1 Bug Tracker

Launchpad's bug tracker tracks all issues submitted to it originating both from users and the crew.

Bug tracker:

<https://bugs.launchpad.net/avaneya>

12.5.3.2 Bazaar Revision Control

We use Bazaar for revision control, hosted on Launchpad. You can browse it on the web here:

³ This explains why top posting is a bad idea: https://secure.wikimedia.org/wikipedia/en/wiki/Top_posting#Top-posting.

<https://code.launchpad.net/~kip/avaneya/trunk>

Some ask why we had not initially chosen a distributed revision control system (DRCS), one class of source control management systems (SCM). SCMs are used to allow multiple people to collaborate over a shared set of files, track revisions and logs, and a number of other things. There are many free programs that allow this, but they can generally be grouped into two categories or paradigms based on how they are expected to be used.

DRCS are akin to peer-to-peer software where they can be used in the absence of a central canonical server. Proponents argue users are better able to work productively when not connected to a network, most operations are much faster since no network is involved, and more. Probably the strongest point raised is it allows participation in projects without requiring permissions from project authorities, and thus arguably better fosters a culture of meritocracy instead of requiring "committer" status. Software that implements these include Mercurial, Git, Bazaar, Monotone, Darcs, and others. This approach has been popularized by the open source movement in recent years, as it captures the "bazaar" approach to software development (think of the Persian marketplace).

CRCS, centralized revision control systems, are akin to peer-to-server model. They have a single canonical repository on a single server. Proponents argue it is more straightforward to contribute to, work is better coordinated, has a more approachable learning curve, backups are more straightforward, and has been around longer. CVS, Subversion, and many others implement this approach. This approach has been popularized by the free software movement, as it captures the "cathedral" approach to software development (think of a central coordinator).

Many people had suggested we use Bazaar because it has a feature that Subversion, what many are use to, does not, DRCS. It is clear that it is perfectly capable of the distributed approach, but that should not be characterized as a feature any more than the specific colour of a car is a feature. It is not a feature, but a preference. Nevertheless, we ended up settling with Bazaar because it can function in the DRCS approach, is the only SCM supported by Launchpad, integrates well with modern graphical GNU desktop environments,⁴ and improves on the features Subversion supports. Besides, it was time to try something new and learning something new is usually not a bad idea.

12.5.3.3 Blueprints

Launchpad provides a feature called Blueprints which allows you to track the design process from basic feature specifications all the way to implementation. Our project's currently available blueprints include:

- **ares-architectural-design:** This document describes the AresEngine's internal architectural design. See the README for details.

<https://blueprints.launchpad.net/avaneya/+spec/ares-architectural-design>

- **ares-package-design:** The AresPackage manifest file format and documentation is provided. See [Section 11.1 \[AresPackages\]](#), [page 22](#) for details.

⁴ As an example, check to see if your distribution carries the *nautilus-bzr* package.

<https://blueprints.launchpad.net/avaneya/+spec/ares-package-design>

- **ares-events-design:** The event type schema and definitions for built-in event types, Events.xsd and Events.xml respectively. See [Section 11.3.7 \[Events\]](#), [page 38](#) for details.

13 Licensing Rationale

Avaneya is composed of different types of files that fall into general categories. The game is licensed under multiple licenses, with a given license for each category. These categories are for *artwork*, *code*, *documentation*, and *music*.

- **Artwork:** This includes all literary works, cinematics, models, audio and voice effects, textures, materials, concept art, fonts, screenshots, and other relevant non-executable creative data.

These objects are licensed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike (BY-NC-SA) 3.0 Unported license, copyright © 2010, 2011 Kshatra Corp. See [Section B.3 \[License CC-BY-NC-SA\]](#), [page 81](#) for the full text of the license.

The license ensures that, under the default conditions, unless given permission, others cannot use it for commercial purposes (NC), must give attribution (BY), and if they alter, transform, or build upon it, they must distribute the resulting work only under the same or similar license to this one (SA). Everyone is free to copy, distribute, and transmit the work, as well as to adapt the work as they like. These are just the default conditions and can be waived with the permission of the copyright holder.

- **Code:** This includes the AresEngine, shaders, Lua scripts, build environment scripts, and any other relevant executable data.

These objects are licensed under the terms of the GNU General Public License 3.0, copyright © 2010, 2011 Kshatra Corp. See [Section B.1 \[License GPL\]](#), [page 63](#) for the full text of the license.

In a nutshell, this license ensures that users have four fundamental freedoms that are always protected. These are the freedom to use the software for any purpose, the freedom to change the software to suit your needs, the freedom to share the software with your friends and neighbors, and the freedom to share the changes you make. The license for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program - to make sure it remains free software for all its users.

- **Documentation:** This includes UML schematics and other design documents, doxygen output, man and info pages, this handbook, and other relevant data.

These objects are licensed under the terms of the GNU Free Documentation License 1.3, copyright © 2010, 2011 Kshatra Corp. See [Section B.2 \[License GFDL\]](#), [page 74](#) for the full text of the license.

The Free Software Foundation explains the purpose of the license as being to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this license preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

- **Music:** This includes all music in Ogg Vorbis, FLAC, Speex, or other formats and associated project files.

These objects are licensed under the terms of their respective artists or publishers. We feel that music does not have to be free, but it is preferred that it be at least shareable. This is the position of Richard Stallman of the Free Software Foundation, along with many artists.

Appendix A Glossary

The following is a collection of terms used in the game world that range from the everyday colloquial to technical terminology. Familiarizing yourself with them will be useful in ensuring the game remains as rich as possible.

Endothermic reaction: A chemical reaction that requires energy to be put into it in order to react. An example would be cooking an egg. The egg doesn't change unless you apply heat to it. This is contrasted with an *exothermic reaction*.

Exothermic reaction: A chemical reaction that releases energy as part of its reaction. An example would be burning wood. As the wood changes, it releases energy in the form of mostly heat.

Genuine Progress Indicator: The GPI is a system of true cost accounting which is intended to be a replacement to the GDP, gross domestic product, sometimes called the gross national product, GNP. The GPI works by taking into account all costs of an activity to society and provides a net balance sheet. The latter, on the other hand, only functions as an income sheet by tabulating the total amount of goods and services produced in a year.

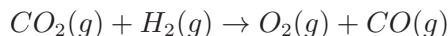
An example where the difference between the two is well illustrated is with the tobacco industry. The GDP would account for the value of all cigarettes sold (a positive figure). Conversely, the GPI would also account for the total value of all units sold, but then go on to subtract the deficits socialized to everyone in the form of health costs, fires, garbage collection, environmental toxicity, and the deaths of approximately 5,000,000¹ people every year (a negative figure).

Regolith: What most refer to as dirt. More technically, it is the the loose heterogeneous mixture of material that blankets the solid rock.

Rhodium: An elemental chemical with the symbol Rh and atomic number 45. It is a member of the platinum family and considered to be the most precious metal of that family, even exceeding the value of gold. It is also one of rarist.

Usually the only way of getting any kind of high quantity mineral is through high-grade ore. This only happens when complex hydrological and volcanic processes have happened, which in our solar system, has only occurred on Mars and Earth - hence why the Moon is barren. But unlike the Earth, Martian deposits have remained untapped.

RWGS reactor: The reverse-water-gas-shift reactor is a method of producing oxygen (O_2) from carbon dioxide (CO_2). This is useful because the latter is plentiful in the Martian atmosphere at 95 %.



¹ *Merchants of Doubt: How a Handful of Scientists Obscured the Truth on Issues from Tobacco Smoke to Global Warming*, Naomi Oreskes, Bloomsbury Press, 2010

The process has been known since the mid 1800s and works by reacting carbon dioxide and hydrogen gasses together over a copper-on-alumina catalyst. Aqua (liquid water) and carbon monoxide gas are produced as byproducts. The aqua is split via electrolysis to produce hydrogen and oxygen gasses. The hydrogen can then be recycled back into the reactor and the carbon monoxide purged out into the atmosphere.

The reactor needs to be at 400 °C and at low pressure. It requires about 180 watts of power, or about 3 m^2 of solar panels on a fully sunny day's average solar flux. At that energy rate, you can expect to produce about 1 kg per day of oxygen, which is sufficient for a single person. The reactor requires power because it is an *endothermic reaction*. However, it is possible to use a *Sebatier reactor* in tandem, which is an exothermic process, to provide the heat required to drive the RWGS reaction.

To start the process, only a small amount of water is required which acts as a reagent. By importing hydrogen from Earth, it acts to the colonists' advantage in allowing it to be leveraged in the creation of water, or hydrogen gas if needed.

Sebatier reactor: A chemical process for creating methane CH_4 from CO_2 and hydrogen. This is useful because carbon dioxide gas is plentiful in the Martian atmosphere at 95 %.



The reactor needs to be at 400 °C and at low pressure. This makes it almost the same as the *RWGS reactor* except that it uses a different catalyst to make methane instead of carbon monoxide. You can either use nickel, which is cheap, or ruthenium-on-alumina, which is safer, but more expensive.

Sol: Short for solar day, the length of time a planet takes to rotate completely on its polar axis with respect to the sun. Terrans call this a day, Martians a sol. See also *yestersol*.

Yestersol: The sol preceding the current one. This is the Mars analogue to the Terran yesterday, but different since the length of a sol on both worlds is different.

Appendix B Licenses

B.1 GNU General Public License

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other

domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

“This License” refers to version 3 of the GNU General Public License.

“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”. “Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular

programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work’s System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users’ Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty

adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a. The work must carry prominent notices stating that you modified it, and giving a relevant date.
- b. The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
- c. You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
- d. If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- a. Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
- b. Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
- c. Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
- d. Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
- e. Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a. Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b. Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

- c. Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d. Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e. Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f. Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party’s predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor’s “contributor version”.

A contributor’s “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor’s essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient’s use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others’ Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work,

but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with

the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

B.2 GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc.

<http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document *free* in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or non-commercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of “copyleft”, which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The “Document”, below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as “you”. You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A “Modified Version” of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A “Secondary Section” is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document’s overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The “Invariant Sections” are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released

under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The “Cover Texts” are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A “Transparent” copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not “Transparent” is called “Opaque”.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The “Title Page” means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, “Title Page” means the text near the most prominent appearance of the work’s title, preceding the beginning of the body of the text.

The “publisher” means any person or entity that distributes copies of the Document to the public.

A section “Entitled XYZ” means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as “Acknowledgements”, “Dedications”, “Endorsements”, or “History”.) To “Preserve the Title” of such a section when you modify the Document means that it remains a section “Entitled XYZ” according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any,

be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their

titles to the list of Invariant Sections in the Modified Version’s license notice. These titles must be distinct from any other section titles.

You may add a section Entitled “Endorsements”, provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled “History” in the various original documents, forming one section Entitled “History”; likewise combine any sections Entitled “Acknowledgements”, and any sections Entitled “Dedications”. You must delete all sections Entitled “Endorsements.”

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an “aggregate” if the copyright resulting from the compilation is not used to limit the legal rights of the compilation’s users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document’s Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled “Acknowledgements”, “Dedications”, or “History”, the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License “or any later version” applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

“Massive Multiauthor Collaboration Site” (or “MMC Site”) means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A “Massive Multiauthor Collaboration” (or “MMC”) contained in the site means any set of copyrightable works thus published on the MMC site.

“CC-BY-SA” means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

“Incorporate” means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is “eligible for relicensing” if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

B.3 Creative Commons Attribution-NonCommercial-ShareAlike

Version 3.0 Unported

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License. b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(g) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License. c. "Distribute" means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership. d. "License Elements" means the following high-level license attributes as selected by Licensor and indicated in the title of this License: Attribution, Noncommercial, ShareAlike. e. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License. f. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or en-

ties who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast. g. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work. h. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation. i. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images. j. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.
3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:
 - a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections; b. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked

"The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified."; c. to Distribute and Publicly Perform the Work including as incorporated in Collections; and, d. to Distribute and Publicly Perform Adaptations.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved, including but not limited to the rights described in Section 4(e).

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

- a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(d), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(d), as requested.
- b. You may Distribute or Publicly Perform an Adaptation only under: (i) the terms of this License; (ii) a later version of this License with the same License Elements as this License; (iii) a Creative Commons jurisdiction license (either this or a later license version) that contains the same License Elements as this License (e.g., Attribution-NonCommercial-ShareAlike 3.0 US) ("Applicable License"). You must include a copy of, or the URI, for Applicable License with every copy of each Adaptation You Distribute or Publicly Perform. You may not offer or impose any terms on the Adaptation that restrict the terms of the Applicable License or the ability of the recipient of the Adaptation to exercise the rights granted to that recipient under the terms of the Applicable License. You must keep intact all notices that refer to the Applicable License and to the disclaimer of warranties with every copy of the Work as included in the Adaptation You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Adaptation, You may not impose any effective technological measures on the Adaptation that restrict the ability of a recipient of the Adaptation from You to exercise the rights granted to that recipient under the terms of the Applicable License. This Section 4(b) applies to the Adaptation as incorporated in a Collection, but this does not require the Collection apart from the Adaptation itself to be made subject to the terms of the Applicable License.
- c. You may not exercise any

of the rights granted to You in Section 3 above in any manner that is primarily intended for or directed toward commercial advantage or private monetary compensation. The exchange of the Work for other copyrighted works by means of digital file-sharing or otherwise shall not be considered to be intended for or directed toward commercial advantage or private monetary compensation, provided there is no payment of any monetary compensation in connection with the exchange of copyrighted works. d. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and, (iv) consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(d) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties. e. For the avoidance of doubt:

i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License if Your exercise of such rights is for a purpose or use which is otherwise than noncommercial as permitted under Section 4(c) and otherwise waives the right to collect royalties through any statutory or compulsory licensing scheme; and, iii. Voluntary License Schemes. The Licensor reserves the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License that is for a purpose or use which is otherwise than noncommercial as permitted under Section 4(c). f. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law,

if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING AND TO THE FULLEST EXTENT PERMITTED BY APPLICABLE LAW, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO THIS EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

6. Termination

- a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.
- b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

7. Miscellaneous

- a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
- b. Each time You Distribute or Publicly

Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License. c. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable. d. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent. e. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You. f. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of this License.

Creative Commons may be contacted at <http://creativecommons.org/>.

Index

(Index is nonexistent)