**Emotional Design Resources**

[**https://www.computer.org/cms/Computer.org/ComputingNow/homepage/2010/0510/T\_IT\_MakingSense.pdf**](https://www.computer.org/cms/Computer.org/ComputingNow/homepage/2010/0510/T_IT_MakingSense.pdf)

[**https://www.w3.org/TR/emotionml/#s5.2**](https://www.w3.org/TR/emotionml/#s5.2)

[**https://alistapart.com/article/personality-in-design**](https://alistapart.com/article/personality-in-design)

[**https://abookapart.com/products/designing-for-emotion**](https://abookapart.com/products/designing-for-emotion)

[**https://gojs.net/latest/samples/decisionTree.html**](https://gojs.net/latest/samples/decisionTree.html)

[**https://stackoverflow.com/questions/8368698/how-to-implement-a-decision-tree-in-javascript-looking-for-a-better-solution-th**](https://stackoverflow.com/questions/8368698/how-to-implement-a-decision-tree-in-javascript-looking-for-a-better-solution-th)

[**http://www.cs.cmu.edu/~yandongl/learningjs/decision-tree-demo.html**](http://www.cs.cmu.edu/~yandongl/learningjs/decision-tree-demo.html)

[**http://corp.yonyx.com/**](http://corp.yonyx.com/) **(decision trees)**

[**http://www.norsys.com**](http://www.norsys.com)

.[**http://www.plexusowls.com/PDFs/using%20bayesian%20networks.pdf**](http://www.plexusowls.com/PDFs/using%20bayesian%20networks.pdf)

[**https://www.analyticsvidhya.com/blog/2016/06/bayesian-statistics-beginners-simple-english/**](https://www.analyticsvidhya.com/blog/2016/06/bayesian-statistics-beginners-simple-english/)

[**https://blogs.sas.com/content/sastraining/2011/01/31/the-bayes-theorem-explained-to-an-above-average-squirrel/**](https://blogs.sas.com/content/sastraining/2011/01/31/the-bayes-theorem-explained-to-an-above-average-squirrel/)

[**https://www.mindtools.com/dectree.html**](https://www.mindtools.com/dectree.html)

[**https://www.countbayesie.com/blog/2015/2/18/bayes-theorem-with-lego**](https://www.countbayesie.com/blog/2015/2/18/bayes-theorem-with-lego)

**GAMING**

[**https://en.wikipedia.org/wiki/Bartle\_taxonomy\_of\_player\_types**](https://en.wikipedia.org/wiki/Bartle_taxonomy_of_player_types)

[**https://uxplanet.org/design-for-emotion-expert-tips-by-aarron-walter-2f847e75a962**](https://uxplanet.org/design-for-emotion-expert-tips-by-aarron-walter-2f847e75a962)

[**https://uxdesign.cc/the-most-overlooked-growth-hack-designing-for-emotions-1a3ba503d4f4**](https://uxdesign.cc/the-most-overlooked-growth-hack-designing-for-emotions-1a3ba503d4f4)

[**https://www.behance.net/gallery/12748107/Weather-Dashboard-Global-Outlook**](https://www.behance.net/gallery/12748107/Weather-Dashboard-Global-Outlook) **(great design)**

[**https://cssauthor.com/web-widget-designs-psd/#Profile**](https://cssauthor.com/web-widget-designs-psd/#Profile) **(widget samples by category)**

[**https://gameanalytics.com/blog/understanding-your-audience-bartle-player-taxonomy.html**](https://gameanalytics.com/blog/understanding-your-audience-bartle-player-taxonomy.html)

**GENERATIVE DESIGN SYSTEMS**

AutoCad

**Notes:**

Myers-Briggs Type Indicator:

* Entroversion €
* Introversion (I)
* Sensing (S)
* Intuition (N)
* Thinking (T)
* Feeling (F)
* Judging (J)
* Perceiving (P)

Decision Network Architecture (DNA)

Uncertainty and complexity are characteristics of human behavior that make it especially difficult to simulate. Uncertainty has largely been ignored in prior behavior models, particularly uncertainty resulting from the natural limitations of perception, especially perception of the intentions of other people. Furthermore, no systematic approach has been proposed to deal with complexity. Our **decision network framework** addresses both issues. Decision networks are a generalization of Bayesian networks [Pea88], also known as probabilistic graphical models, which combine probability theory and graph theory to capture uncertain knowledge in a natural and efficient manner. An attractive feature of the decision network is that it is a powerful tool for modeling decision making under uncertainty.

Ball and Breese [BB00] encoded emotions and personality using Bayesian networks

Closer to our approach, Ball and Breese [BB00] encoded emotions and personality using Bayesian networks. Unlike our work, however, their emphasis was on conversational agents with speech recognition and generation. Kshirsagar and Thalmann [Ksh02] also used a Bayesian network to model personality and mood in a chat application, as did Egges et al. [EZKT03] to model mood in their conversational agent. The work closest to ours is that by Hy et al. [HABL04] who simulated simple behaviors for a firstperson shooter game character by using a Bayesian network to specify finite-state-machine-like behavior selection, and to learn by imitating a human player

Although prior animation work and existing computer game titles have used finite state machines, fuzzy logic, neural nets, scripting, smart environments, and Bayesian networks, to our knowledge, ours is the first effort in computer graphics to develop and demonstrate a unified framework for behavioral animation based on decision networks. The decision network (or “influence diagram”), which was introduced by Howard and Matheson [HM81] in the area of decision analysis, combines probability theory and utility theory to provide a simple visual representation of a decision problem. Decision networks extend Bayesian networks by adding actions and utilities.

Decision making mechanisms:

* Fuzzy logic
* Neural networks
* Rule-based architectures
* Decision networks

display the essential elements of the problem, including objectives, uncertainties, interpretations, and decisions, and how they influence each other, as well as the clear attribution of outcomes to the inputs that generated them.

4.1. Decision Network Framework A complex human behavior usually requires a sequence of assessments and decisions to be made. To model the behavior, the various contributing factors as well as their interrelationships must be identified, specified, and quantified. A decision network encodes events, represented by nodal variables, and causal relationships between them, represented by directed edges. This facilitates behavioral modeling as it is natural to think of behaviors as causal relations between events. A decision network computes rational decisions based on what the agent wants and what it believes, whereas a purely logical agent would not be able to handle uncertainty combined with conflicting goals [RN03]. We take advantage of the features of decision networks that address the key issues identified in Section 1. The uncertainties associated with various variables of interest are represented by the probability distributions encoded in the decision network. The decision network’s powerful inferencing capability enables the explanation of observations made about the world, as well as predictions based on the evidence. The use of decision networks provides a convenient way to control how the character makes decisions. Adjusting the conditional probabilities and the utility functions will influence how the decision gets made. Another way to exert control is to adjust internal parameters, which will be monitored at simulation time by the network in order to make inferences and assessments. In our application, not every decision need take all sensory and internal factors into consideration. To avoid the potential intractability of large decision networks, we build the behavioral model as a hierarchical set of relatively small decision networks. The hierarchical structure helps to ensure a manageable number of variables in each individual network and reusability of the functionality of component networks. At the lower level, a separate, smaller decision network structure is implemented for each decision item, while at higher level(s) the decision network structure at each node represents how a decision is made based on results from its children nodes. Decision network behavior models are also readily extensible, as we will demonstrate in Section 4.2.4. Fortunately, there is no need to reinvent the wheel when applying decision networks. We use **Netica (www.norsys.com),** which is a commercial-quality implementation of Bayesian and decision networks along with a convenient GUI. It uses the junction tree algorithm to evaluate the networks and draw inferences. † Applying our decision network framework, we have developed a set of networks that implement each of the four behavior models. Each network is responsible for drawing an inference or making a sub-decision that contributes to the final behavior. They are invoked only when there is a need to make the corresponding inference or decision.

The role played by Netica, and DI-Guy for that matter, in our work is no different than the role played by, say, Matlab and Maya in many CG research projects. Netica serves simply as a software tool.

Modular and hierarachical

From: <http://www.norsys.com/belief.html>

## Belief Networks and Decision Networks

Belief networks (also known as Bayesian networks, Bayes networks and causal probabilistic networks), provide a method to represent relationships between propositions or variables, even if the relationships involve uncertainty, unpredictability or imprecision. They may be [**learned automatically**](http://www.norsys.com/WebHelp/NETICA/X_Neticas_Learning.htm) from data files, created by an expert, or developed by a combination of the two. They capture knowledge in a modular form that can be transported from one situation to another; it is a form people can understand, and which allows a clear visualization of the relationships involved.

By adding decision variables (things that can be controlled), and utility variables (things we want to optimize) to the relationships of a belief network, a decision network (also known as an influence diagram) is formed. This can be used to find optimal decisions, control systems, or plans.   
For examples of a variety of Bayes nets, explore our [**BN library**](http://www.norsys.com/netlibrary/index.htm).

Norsys specializes in making advanced belief network and decision network technology practical and affordable.

To try it for free: [**download**](http://www.norsys.com/download.html) the latest version, leave the password dialog box empty and click '**Limited Mode**'.

Bay-ee-see-en

**1.4.1 Probabilities need not be exact to be useful**

Some people have shied away from using Bayes nets because they imagine they will only work well, if the probabilities upon which they are based are exact. This is not true. It turns out very often that approximate probabilities, even subjective ones that are guessed at, give very good results. Bayes nets are generally quite robust to imperfect knowledge. Often the combination of several strands of imperfect knowledge can allow us to make surprisingly strong conclusions.

**1.4.2 Causal Conditional Probabilities are easier to estimate than the reverse**

Studies have shown people are better at estimating probabilities "in the forward direction". For example, doctors are quite good at giving the probability estimates for "if the patient has lung cancer, what are the chances their X-ray will be abnormal?", rather than the reverse, "if the X-ray is abnormal, what are the chances of lung cancer being the cause?" ([Jensen96](http://www.norsys.com/tutorials/netica/tut_refs.htm#Jensen96))

Bayesian analysis uses prior information plus data to arrive at predictions that are expressed in terms of posterior probabilities.

Dynamic vs. static analysis

Conditional probabilities using bayes theorm and Bayesian analsysis

Bayes brings into play modifiers…information that influences the ultimate outcome (decision). It goes beyond a simple static model based, lets says on the simple probability that “killers” would respond to a gaming stratedgy.

Certainly age would come into play, younger workers might have a strong inclination to gaming..but what about orlder workers…or males vs. females. What about IQ? And what about all the various ways to incorporate a game stategy into the software design? Factor in artifacts decisions and design features (leaderboards? Rankings?)

We should build sophisticated decision network architecture (DNA) into our design system from the get go (Baysian) even if we start off making relatively simple UI decisions. I think we need to plan for an architecture that recognizes that a static DNA only goes so far when dealing with emotion, which requires a dynamic model.

extracted reason out of intuition.

From: https://ac.els-cdn.com/S2351978915009646/1-s2.0-S2351978915009646-main.pdf?\_tid=19c49b08-1264-11e8-9f46-00000aab0f6b&acdnat=1518708357\_72ea22bb772d825b67068934b71a3a42

Leveraging personality models provides an effective means to achieve these goals.By creating a framework of personality factors which highlights scientifically-vetted, culturally-sensitive factors, marketeers increase their understanding of a market segment’s mental state and underlying motivations, and therefore can improve the effectiveness of persuasive messages targeting that market.

PAST supports effective message creation by mapping the target market segment personality profile to appeals, techniques, and dissemination strategies that maximize message impact based on applying what will most resonate with the market segment’s characteristics. Each part of PAST within the marketing workflow leverages relevant literature from academia to enable marketeers to effectively conduct cross-cultural analyses in dynamic operational environments while mitigating potential biases. At the time of writing, we are implementing PAST in software and will be able to report on preliminary user studies in a future publication.

*Giving users the power to choose changes the tone of their response. When forced to change, people often react negatively. Allow people to change on their own schedule, and you empower them, diffusing animosity. We’d all rather hear “You may …” instead of “You must ….”*

*\*\*\**

*Surprise, delight, anticipation, elevating perceived status, and limiting access to elicit a feeling of exclusivity can all be effective in getting your audience to fall in love with your brand. But your tactics must be appropriate for your audience and brand experience.*

*Great content delivered in an emotionally engaging manner is like kryptonite for apathy.*

**Virtual Check Ride and Station Manager**

First iteration will be for Managers (6 of them) and one Supervisor Manager.

How personality and **attitude** influence design decisions:

* Content strategy
* Content presentation style
* Writing style and vocabulary choices, sentence length and amount of text
* Images, number of, size and subject matter
* Interactive elements, an interface that is ‘alive’
* Level of in app support and guidance, hand holding and error resolution
* Environment, light/dark interface, font-size, colors, contrast.
* Performance issues, is the app responsive and perceived to be fast, not to frustrate.
* Level of explanation required to complete a task and use the software.
* The right amount of text for any given user type, the right length and vocabulary
* The way content is presented, abstracted, storytelling, step by step, etc.
* Lots of options vs fewer options
* Lots of content vs sparse content (just the facts, no non sense)
* Animations and video
* Social opportunities
* Things vs. people in content including images.
* Tell me how to do it and let me do it
* Hold my hand and lead me step by step
* Let me figure things out for myself
* Its important that I be able to add my personal touch, lots of preference options
* I hate to set things up, I’m fine with the default.
* I don’t like a clean, antiseptic screen. I like something warmer and with personality.
* It turns me off when the software is all business. Give me something to smile about.
* I really like to feel connected to my group and coworkers.
* I think they really just want to spy on me and make my life miserable, and this software is the way they can do that.
* I like lots of shortcuts
* I like to hack software and figure out how to do things developers never thought about.
* I don’t mind if they spy on me because I have nothing to hide. I am the best and like it that way.

**Bartle Taxonomy**

**Explorer Traits and Attitudes**

1. Focus
2. Preparedness
3. Conviction
4. Perseverance
5. Creativity
6. Curiosity
7. Resilience
8. Risk taking
9. Independence
10. A sense of higher purpose

Personality Traits:

* Thrill seeker
* Don’t ask me to doggedly execute routine tasks
* I thrive on a steady stream of new people and place and adventures
* I like novelty
* High energy
* Fast paced environment
* Fun, charsimatic and generous with their time
* The loud, humorous person in command of the corner table
* Just opened the second bottle of wine.
* Its easy to overlook their flaws.
* Independent and often unconventional
* Not joiners of group think
* Few commitments to get in the way of spontaneity.
* I like a lot of content and resources
* I like to read, bring on the text
* Clean design if ok, but I like lots of bits and pieces of content, it make me feel like there’s a lot of good stuff to find. I get excited to explore when it looks like there’s a lot of good content.
* I like in depth content too, I don’t want a bunch of summaries, unless you can expand it out for details.
* I love dashboards, metrics and analytics.
* I love charts and tables and detailed numbers.
* I want to learn as much as I can.
* I am text oriented.
* Videos are ok but I like to read.
* I am visual.
* I like process, the more complex the better.
* I like to figure things out by myself.
* Im happiest when I’ve explored everything possible about any given subject.
* I like to become an expert.
* I like the process of becoming an expert.
* I get board easily
* I enjoy getting hold of a new piece of software and figuring it out.

Gamification & real-time feedback

Strategy, tactics, features, tools and generic component UI building blocks (widgets)

* KPI’s
* Real time feedback
* Bet on yourself
* Personalized learning

**“Building widgets explorers love…”**

*Widgets inspired by what motivates explorers: their attitudes and emotional triggers*

**Explorer Profile**

Give an explorer a rich ecosystem in which to roam and explore. They like content and lots of it. Don’t ask them to join in the group effort, because they are a team of one. Build a very personal ecology that caters to this unconventional iconoclast. They enjoy complexity and please don’t spare the details. Few commitments and the hell with plans, they like things spontaneous. Exploring is all about discovery and if it’s risky, so much the better.

**Explorer Widgets**

* **The Rear View Mirror** (what I need to know about yesterday) – interactive timeline
* **Task Robot** - (Automate a repetitive tasks because explorers hate routine tasks)
* **Future Cast**– (RSS feed domain trends and news) *“I’m easily bored so keep it fresh.”*
* **Scorecard -** (visual kpi’s)
* **Peer to Peer** (Sharing lessons learned) *“One word: crowdsourcing.”*
* **Something New** (Learning center*) “Because what I don’t know can kill me.”*
* **Right This Minute (**screen ticker) *“High energy and fast paced.”*
* **Need to Know** (A visual, highly interactive dashboard) *“My favorite verb is drill down”*
* **On Track** (A GPS viewer)

What we need to do is pick 2 or 3 superstar widgets and perfect them. Test and iterate.

Ask…what can we do to help explorers like to use our software.

They need to deliver results and that means they need to motivate their drivers to do better.

Widget: Tip of the day…daily management tip (learn to manage better)

Widget: My drivers (driver profiles for managers to get to know them better. What motivates them)