Murlin

"**M**y **url**'s **in** one place"

an **overlair** project

Graphical access to a personal database

"The Greeks had made an art of it...the idea of using *loci* and *imagines agentes* (places and active vivid images) for **dynamic, active storage** of *res* and *verba* (things and words)"

Volker Grassmuck

"The universe (which others call the Library) is composed of an indefinite and perhaps infinite number of...galleries"

Jorge Luis Borges

Table of Contents

Purpose

Extending Research

Personal Motivation

> “We see a thriving software industry that largely ignores research, and a research community that writes papers rather than software.”
>
> **Rob Pike**

Extending Research

> "Systems research cannot be just science;
> there must be engineering, design, and art."
> **Rob Pike**

In an pseudo-academic way, part of the motivation for this project is to extend the boundaries of research into the field of human augmentation technology. Evolutions in technology over the last century has led to a Universal Material, a combination of software and hardware, capable of being molded arbitrarily. As processing power and usable memory have increased, so too have the dimensions over the types of forms we can create with this material.

Much of the research into this field, undertaken by thinkers like Bush, Engelbart and Kay, was done before modern advances in graphics, processors, and memory. Their visions were mainly speculative, designing for material they did not yet possess, though they understood that one day we would have such material. I'm not sure if such a day has come to pass, but I can't help but feel, in is this digital era, that we possess more than suitable material to explore their designs more wholeheartedly. This project's aim is to surface effective means of storing, representing and manipulating *digital material* by investigating the historical record of innovation and considering our own natures and the nature of the *digital mediums*.

The ultimate goal of this work is to improve our ability to represent mental images with digital forms.

Personal Motivation

“What I cannot create, I do not understand”
Richard Feynman

In a personal way, this project is motivated by my desire to understand the world around me. Being a product of the Digital Era, I have been born into and will live in the midst of a revolution in representation—a digital-first ethos has swept over the world and will pervade for the remainder of my life. As aspects of my work and play continue to centralize around the digital context, it has become increasingly paramount to gain ownership over this context.

Part of my understanding process is in the creation of software tools and musical pieces. These activities involve the study of material: documentation, prior art, expert opinion; as well as the design of new material, from naïve sketches to matured forms. In this pursuit, it is clear that I am lacking for tools, and spread too thin between disconnected digital contexts.

I need a “super-structure”—a connected set of digital entities that provides linkages over the different aspects, dimensions and stages of my projects, as well as linkages between these projects themselves. It is my vision that in the creation of such digital structures, I will gain insight on the abstraction levels, patterns, and connections latent in my work, and find useful visual representations for these structures.

Prior Art

What has been will be again,
what has been done will be done again;
*there is **nothing new** under the sun.*
Ecclesiastes 1:9

Creating is **remixing**. To a large extent, *new ideas are old ideas* in new combinations.
Bret Victor

Most ideas come from **previous** ideas.
Alan Kay

NLS

HyperCard

Boxer

Xanadu

SmallTalk

Pad++

NLS

by Doug Engelbart

"An Augmented Knowledge Workshop"

"The **basic conceptual unit** in NLS, at each node of the hierarchical file, is called a "**statement**" and is usually a paragraph, sentence, equation, or other unit that one wants to manipulate as a whole."

"There is a convention (called a "**link**") for **citing documents** that allows the user to specify a **particular file, statement within the file and view specification** for initial display when arriving in the cited file."

"Not just hypertext, but graphics, multiple panes, efficient navigation and command input, interactive collaborative work, etc. An **entire conceptual world and world view**"

"We are very much concerned with the speed and flexibility with which one form can be transformed into another, and with which new material can be located and portrayed."

Principle 1:
Object as Primitive

Principle 2:
Link as Primitive

Principle 3:
World Creation

Principle 4:
Multiple Representations

Xanadu

by Ted Nelson

“Parallel Pages, Visibly Connected”

“Every document is **uniquely and securely identified”**

“What we need is not just **visible connection**, but a rational and extensible structure behind it-- a way of grouping parallel documents and managing their visible interconnections”

“Every document can contain links of any type including virtual copies (**"transclusions"**) to **any other document in the system** accessible to its owner.”

“HTML is precisely what we were trying to *prevent* -- ever-breaking links, links going outward only, quotes you can't follow to their origins, no version management, no rights management.”

Principle 1:
Object as Primitive

Principle 2:
Link as Primitive

Principle 5:
Composition by Transclusion

Principle 6:
Play Nicely with Outside World

HyperCard

by Bill Atkinson

"A tool for making tools"

"A database with **simple form layout**, **flexible support for graphics**, and **ease of programming**"

"A **software erector set**, allowing *non-programmers* to put together **interactive information**"

"A much faster way to establish a **relationship** between ideas, facts, theories and thoughts."

Principle 7:
Personal Database

Principle 8:
Accessible

Principle 2:
Link as Primitive

SmallTalk

by Alan Kay, Dan Ingalls and co.

“A philosophy of decomposition in the form of a programming language”
Bret Victor

“A computer language should support the concept of "object" and provide **a uniform means** for referring to the objects in its **universe**.”

Principle 1:
Object as Primitive

“If a system is to serve **the creative spirit**, it must be entirely **comprehensible to a single individual.**”

Principle 8:
Accessible

“The design of a language for using computers must deal with internal models, external media, and the **interaction between these** in both the human and the computer”.

Principle 6:
Play Nicely with Outside World

Boxer

“Naturally expressed hierarchical connections”

by Andrea diSessa and Harold Abelson

Principle 3:
World Creation

A **medium** for **microworlds**

Principle 1:
Object as Primitive

All computational objects are represented in terms of boxes, which are regions on the screen that contain text, graphics, or other boxes. Boxes within boxes represent hierarchical structures.

Principle 8:
Accessible

A way for **non-experts** to control a reconstructible medium, much like written language, but with dramatically extended interactive capabilities.

Principle 9:
Extensible

One major benefit of programmability is that even professionally produced items become **changeable, adaptable, fragmentable, and quotable** in ways that present software is not...a reconstructible medium should allow people to build **personalized computational tools** and **easily modify tools** they have gotten from others.

Pad++

"general-purpose substrate for exploring visualizations of graphical data with a zooming interface"

by Ken Perlin, Ben Bederson, and co.

"We envision a **much richer world** of **dynamic persistent informational entities** that operate according to multiple physics specifically designed to provide cognitively facile access. The physics need to be designed to **exploit semantic relationships explicit and implicit in information intensive tasks** and in our interaction with these new kinds of computationally-based work materials."

"It is natural to see the details of an object when **zoomed in** and viewing it up close. When **zoomed out**, however, instead of simply seeing a scaled down version of the object, it is potentially more effective to see a different representation of it."

Principle 3:
World Creation

Principle 7:
Personal Database

Principle 2:
Link as Primitive

Principle 4:
Multiple Representations

← ⌂

… and more

Apps like Roam, Notion, Muse, Kosmik, Kinopio, RemNote, Obsidian, Clew, Clover, Athens, Craft, and others are currently developing solutions within this space, reflecting many of the principles revealed above by their historical ancestors. As these designs evolve, we at overlair will be following along curiously.

Principles

1. Object as Primitive

2. Link as Primitive

3. World Creation

4. Multiple Representations

5. Composition by Transclusion

6. Play Nicely with Outside World

7. Personal Database

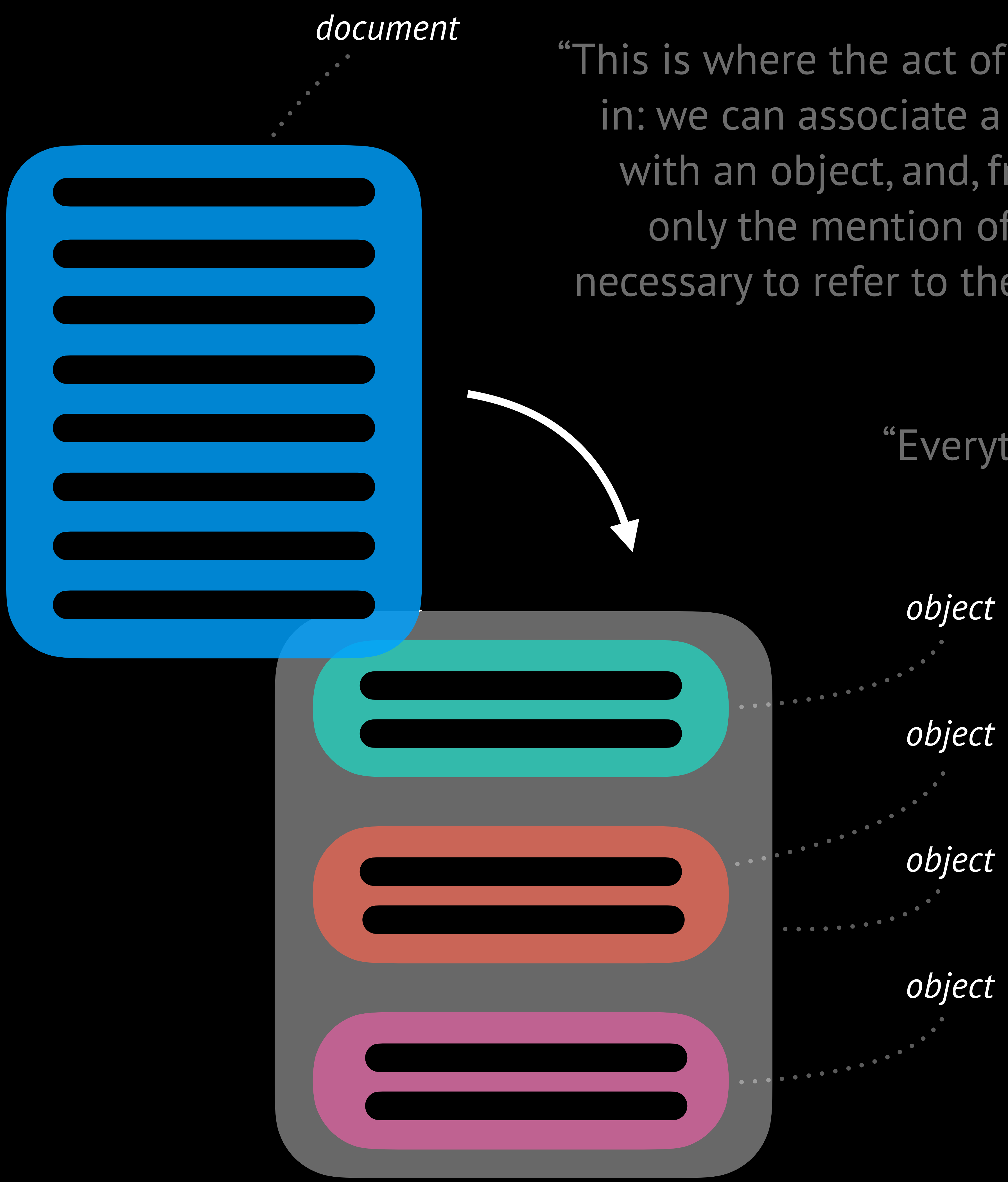
8. Accessible

9. Extensible

Identified Objects as Primitives

In order to have a **unified interface** over concepts, we need to have a **singular notion** of these "objects", represented in the computer as a data model and graphical interface. This pattern occurs repeatedly in the research— one core data model, often recursively nestable, which acts as a "node" within larger **trees** or **graphs**. By having a singular core "type", we can compose these larger networks from units that are nestable and connectable.

We convert from considering our digital universes as "files" and "documents", but as "objects", "blocks" or "units" that we compose structures with.



> "This is where the act of reference comes in: we can associate a unique identifier with an object, and, from that time on, only the mention of that identifier is necessary to refer to the original object."
>
> **Dan Ingalls**

> "Everything is an object"
>
> **Alan Kay**

Principle 1

Connect Objects using Links

Human minds work by forming associations, analogies, metaphors and connections between concepts. These linkages provide reference for relationships, tracing back thoughts, and providing pointers to "mental places".

Digitally, we can think of *hyperbundles,* formed from interconnected hyperlinks, creating a network of links between digital "places" or addresses. These links allow us to form a conceptual network of digital content, that is also supremely navigable. Just as our mind uses associations to build "trails" to follow, we can build these using hyperlinks.

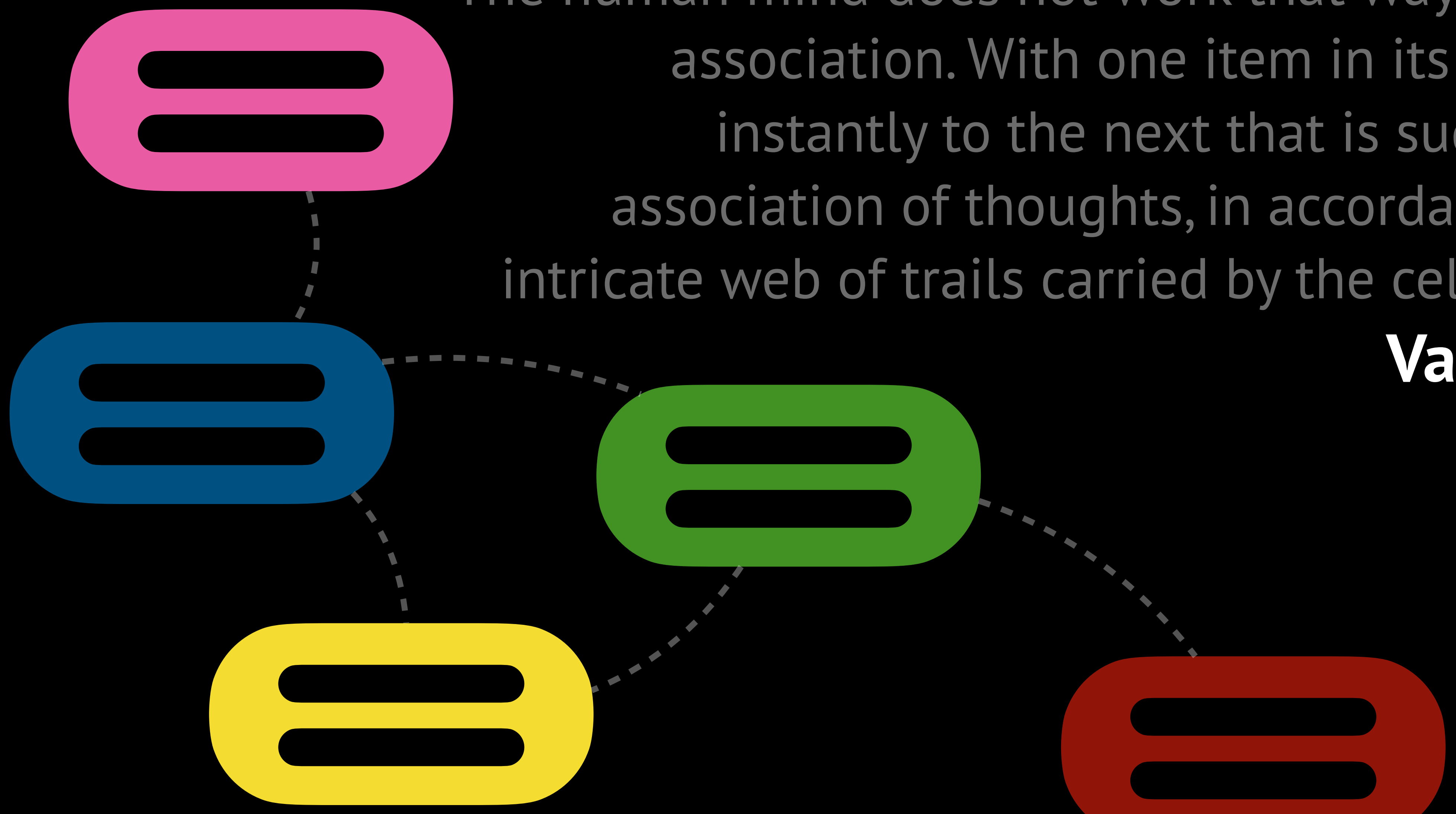
These networks serve not only our own individual purposes as systems of references, allowing the construction of abstractions, connections and collections; but also to share with others, in collective ways—we can share representations in a universal material, accessible through the universal medium of the Internet.

Humans rose above the lower forms of life by evolving the biological capability for developing abstractions and concepts. They could manipulate these concepts within their minds to a certain extent, and think' about situations in the abstract. Their mental capabilities allowed them to develop general concepts from specific instances, predict specific instances from general concepts, associate concepts, remember them, etc

Douglas Engelbart

The human mind does not work that way. It operates by association. With one item in its grasp, it snaps instantly to the next that is suggested by the association of thoughts, in accordance with some intricate web of trails carried by the cells of the brain.

Vannevar Bush



Principle 2

Creating Digital Worlds

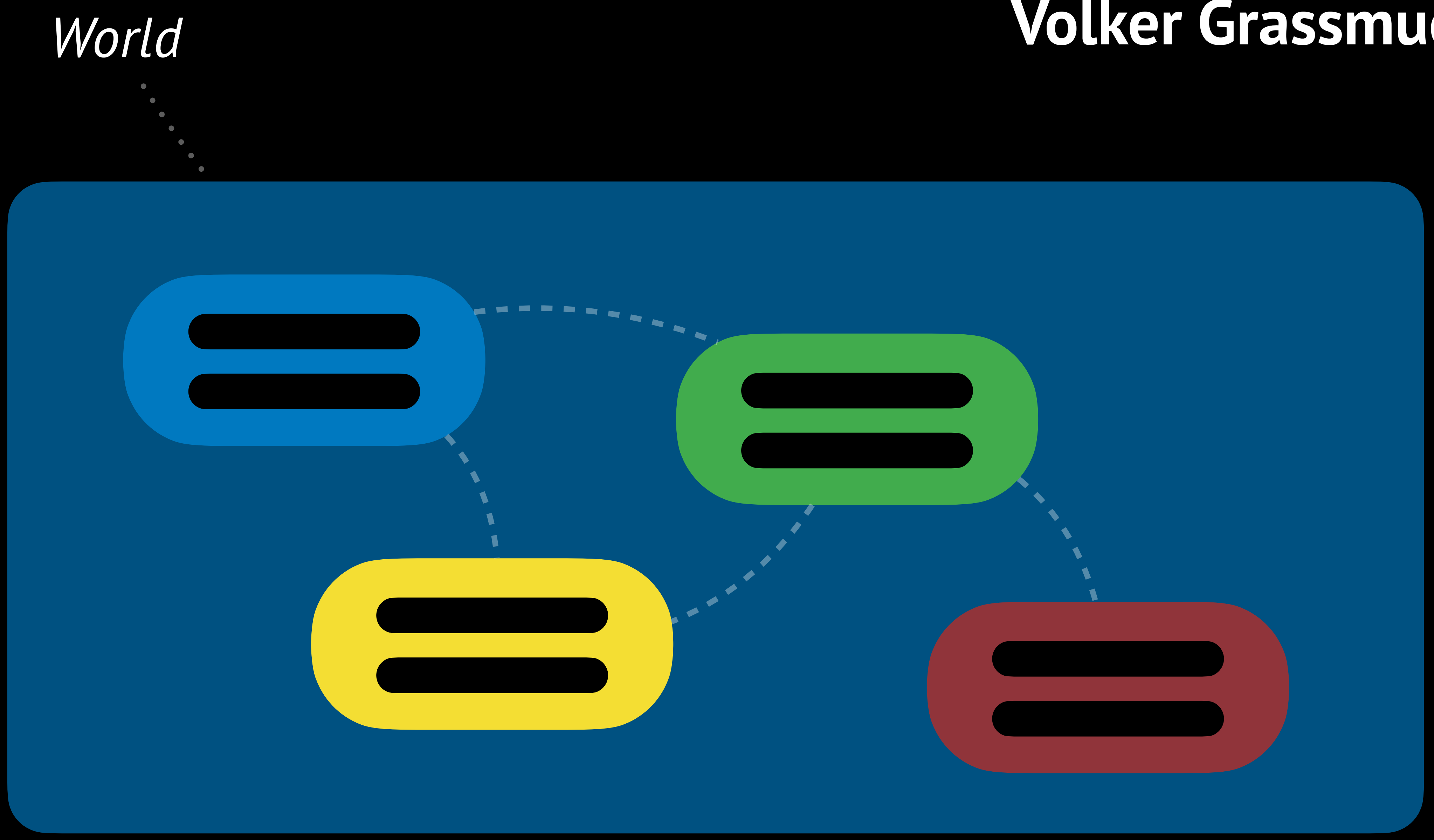
It is not enough to just store a network of objects— we need ways to interact and interface with these objects, and digital spaces in which to do so. The advantages of these digital spaces is the moldable, flexible and infinite properties of digital material. While the screens we view these worlds through are of fixed size, we can represent arbitrarily-sized spaces, and interact with objects in these.

Here, the advantages of hyperlinks, the ability to teleport to another location in hyperspace, shine bright. The digital spaces we build, filled with hyperlinked digital objects, have divorced notions of time and space.

The construction of this world is in part a result of the backing data structures, but manifests most strongly in the interface through which it is interacted with. It is the development of the this interface that is key to the expressions of digital worlds.

> "I decided that Sketchpad's notion of a general window that viewed a **larger virtual world** was a better idea than restricted horizontal panes"
>
> **Alan Kay**

> The interface is the visible 'tangible' surface of media, the nexus point or gateway between man and media, i.e. **the world.**
>
> **Volker Grassmuck**



Principle 3

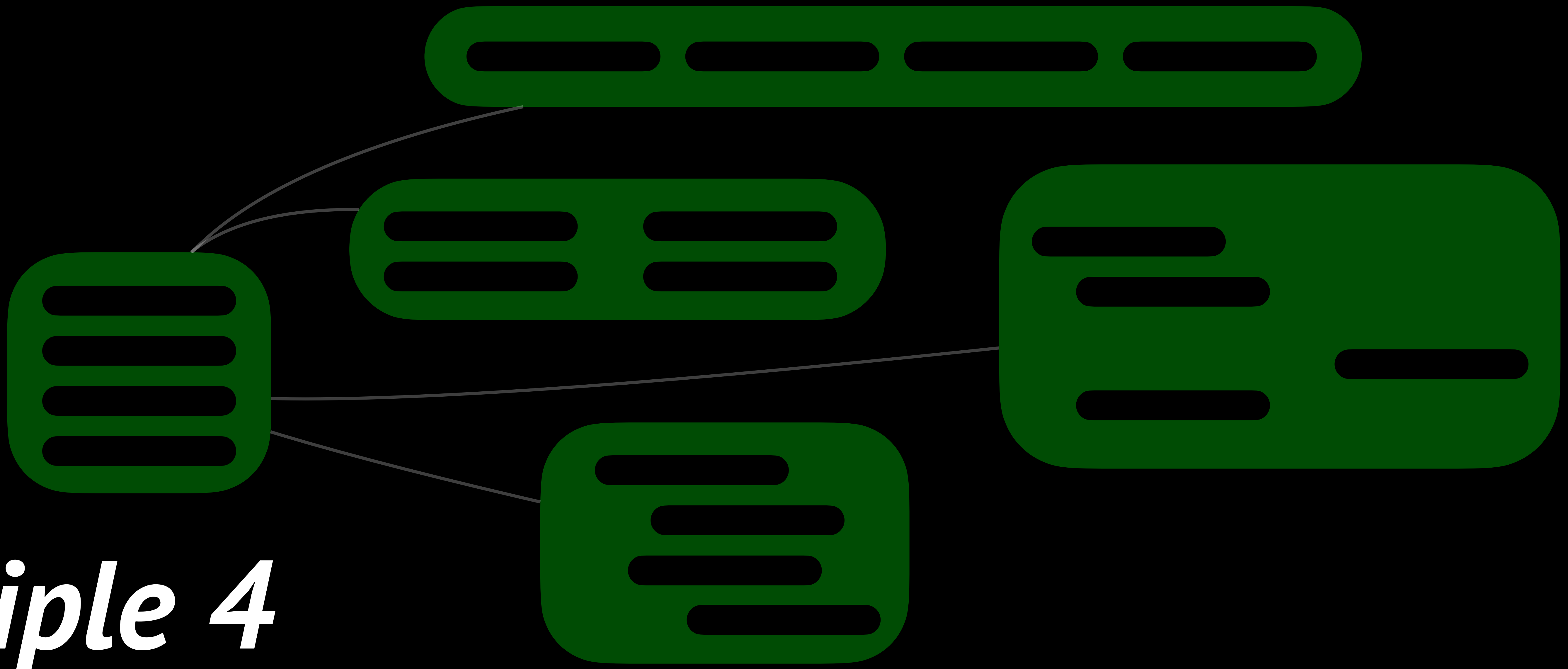
Multiple Representations

The advantage of digital material is that the content and style are separated—we store a data representation, and display a graphical representation. This modularity allows us to create multiple graphical representations over a single data model.

Further, by understanding our data in terms of "types", we can develop mappings between graphical forms and data types, allowing arbitrary data conforming to a certain to assume known graphical layouts and styles.

> "A given structure of concepts can be represented by any of an infinite number of different symbol structures, some of which would be much better than others for enabling the human perceptual and cognitive apparatus to search out and comprehend the conceptual matter of significance and/or interest to the human."
>
> **Doug Engelbart**

> "Essence is abstract, in that the conceptual construct is the same under many different representations"
>
> **Frederick Brooks**



Principle 4

Composition by Transclusion

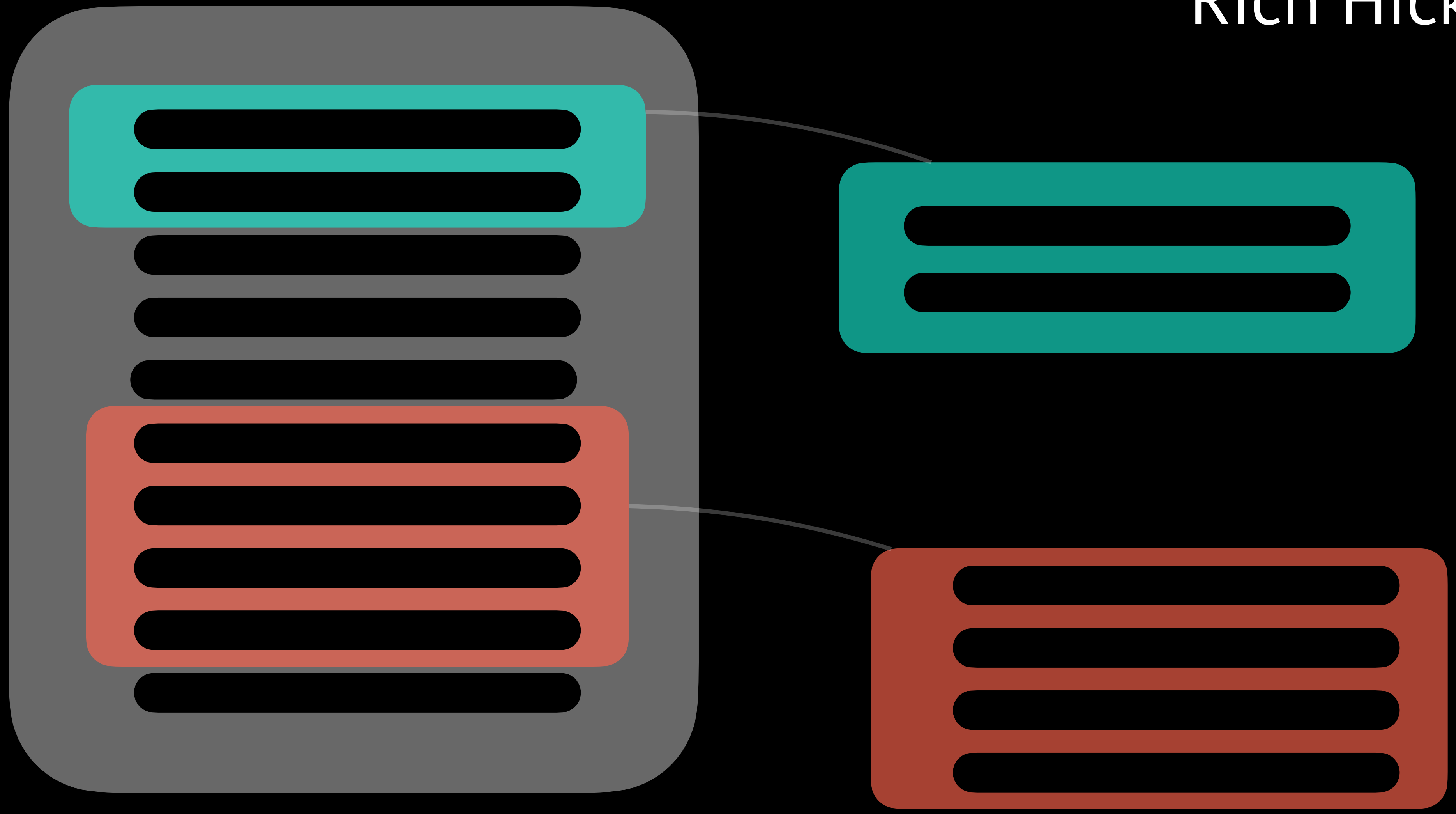
It is not enough to just build compositions of objects, to create upwards abstractions levels from groups of objects (collections), we also need to be able to express linkages between subsections within the data stored by documents, whether it is a media item or set of objects (selections).

This ability to express "lower-level" objects within your universe of objects is essential in balancing the directionality of growth, adding the downward component of decomposition to a system that also supports abstraction and composition in the upward direction.

These linkages also allow for annotation over subsections of content, as well as composition of these subsections into higher-order objects, so we can build "collages" of content from other objects, each linkage expressing a reference and allow navigation to, or display of, the referenced object.

"Modularity is the human mind's lever against complexity. Breaking down a complex thing into understandable chunks is essential for understanding, perhaps *the essence* of understanding."
Bret Victor

"Design is separating into **things that can be composed.**"
Rich Hickey



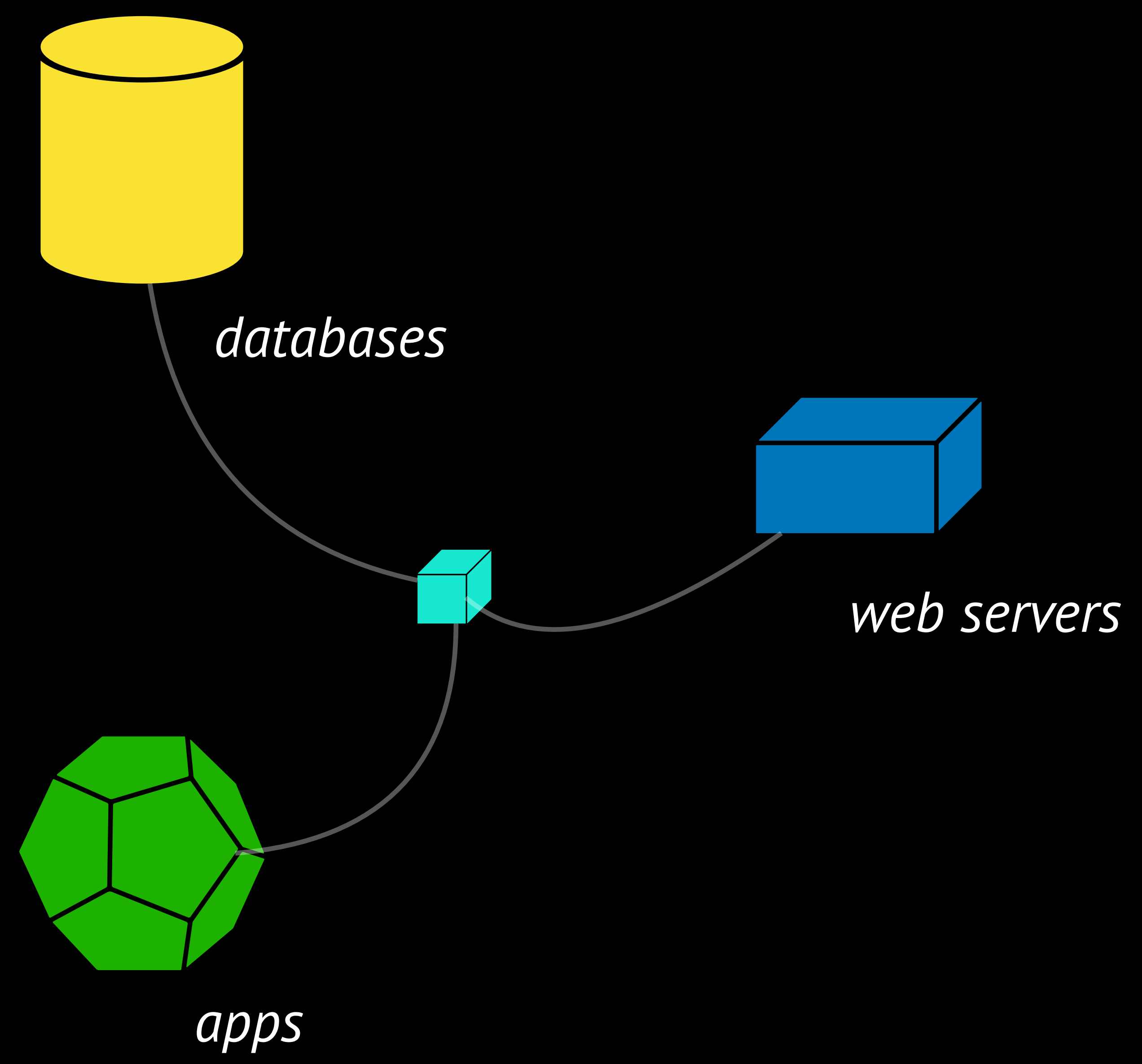
Principle 5

Plays Nicely with Outside World

“The limits of my language are the limits of my world”
Ludwig Wittgenstein

In order to take advantage of the potential within a single computer, we need to allow it to connect to networks of other computers. This involves both allowing us to connect to other users, sharing data when we establish trust, or to the wider Internet of content, allowing connection with data stored on arbitrary public-facing computer, or server.

We can view this world as part of any networking-enabled software’s world, and thus, the extent to which we can interface with this wider world determines the dimensions of the software’s world. In constructing networks of content, we see the value in being able to interface with more content, as there is more surface area to attach concepts to, or draw concepts from.



Principle 6

Personal Database

A common thread amongst most of the projects I've read about is the development of a personal database that provides not only storage, but also ways to consistently display, search, filter and sort through the information.

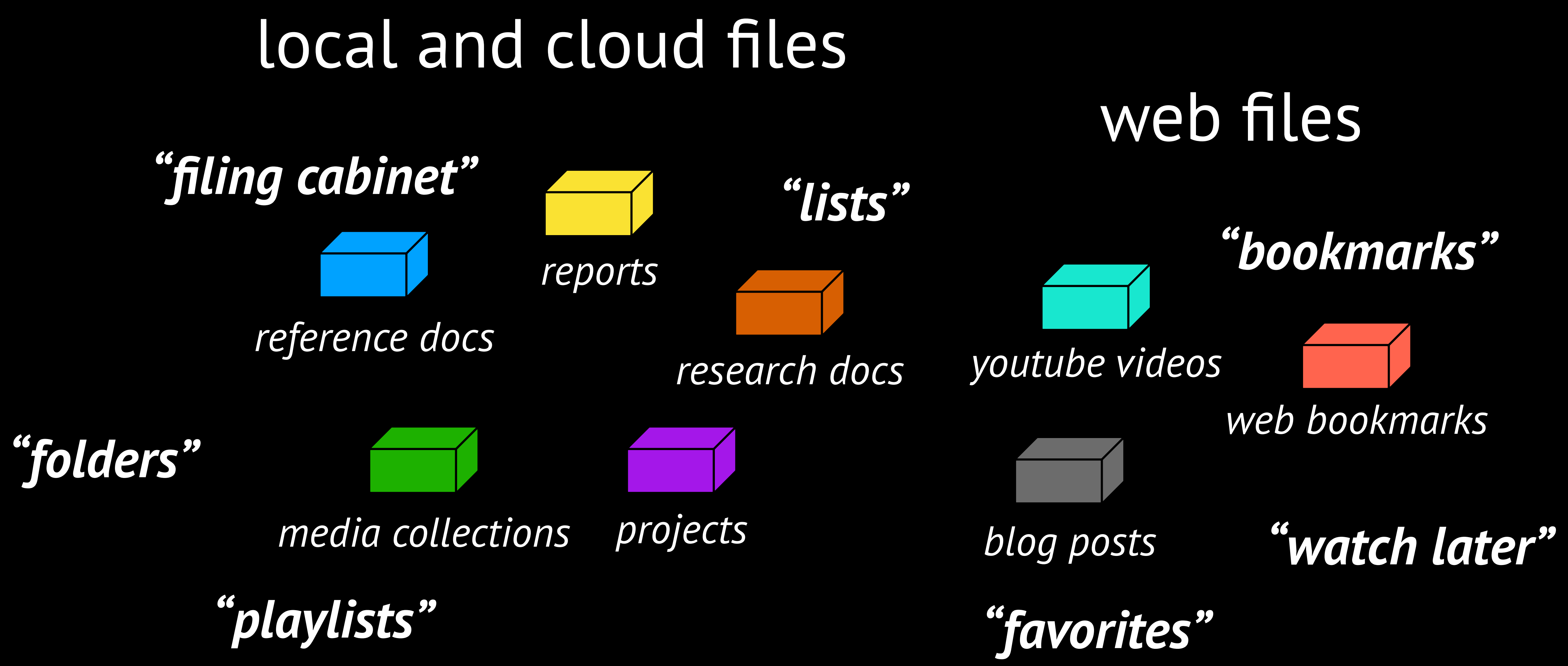
It is the models in this database, often represented in data as a "graph of trees", in which we express both hierarchical and direct links, forming networks of objects. Often, the type of content these objects hold is variable, allowing text, image, graphic, or other computer-recognizable data-format to be stored (or referenced) by the database, and displayed and interacted with in the interface.

We can think of most of these projects as developing "skins over data models", providing the graphical components to display different data types, and laying these out onto documents or canvases, *worlds* which the use to display *places* within one's universe of objects.

Principle 7

> **Cesare Pavese observed: to know the world we must construct it.** In other words, *we make not just to have, but to know.* But the having can happen without most of the knowing taking place
>
> **Alan Kay**

> What we have done in the development of our augmentation means is to construct a superstructure that is a synthetic extension of the natural structure upon which it is built.
>
> **Doug Engelbart**



local and cloud files

web files

"filing cabinet"

reference docs

reports

"lists"

research docs

"folders"

media collections

projects

"playlists"

"bookmarks"

youtube videos

web bookmarks

blog posts

"watch later"

"favorites"

Accessible

It is not always desirable to present a complex set of interactions, or a crowded interface full of controls. We want to contain the complexity to the extent we can—the symbols and concepts we are working with are complex enough!

In simplifying the way we interface with a system, we allow for wider usage, and for a lower barrier to entry. There is a balance here—we can overcompensate and simplify to the point where the system loses it’s expressable power.

One way to simplify and make more accessible, without losing any amount of controls, is to make the interactions and metaphors more *natural,* such that there is an instinctual nature to the interactions that occur.

> In the long haul the person with a simpler system is gonna wipe the plate with you
>
> **Rich Hickey**

> An essential quality necessary for a truly popular medium: the possibility for personal construction by users at all levels of competence
>
> **Hal Abelson/Andrea diSessa**

Principle 8

Extensible

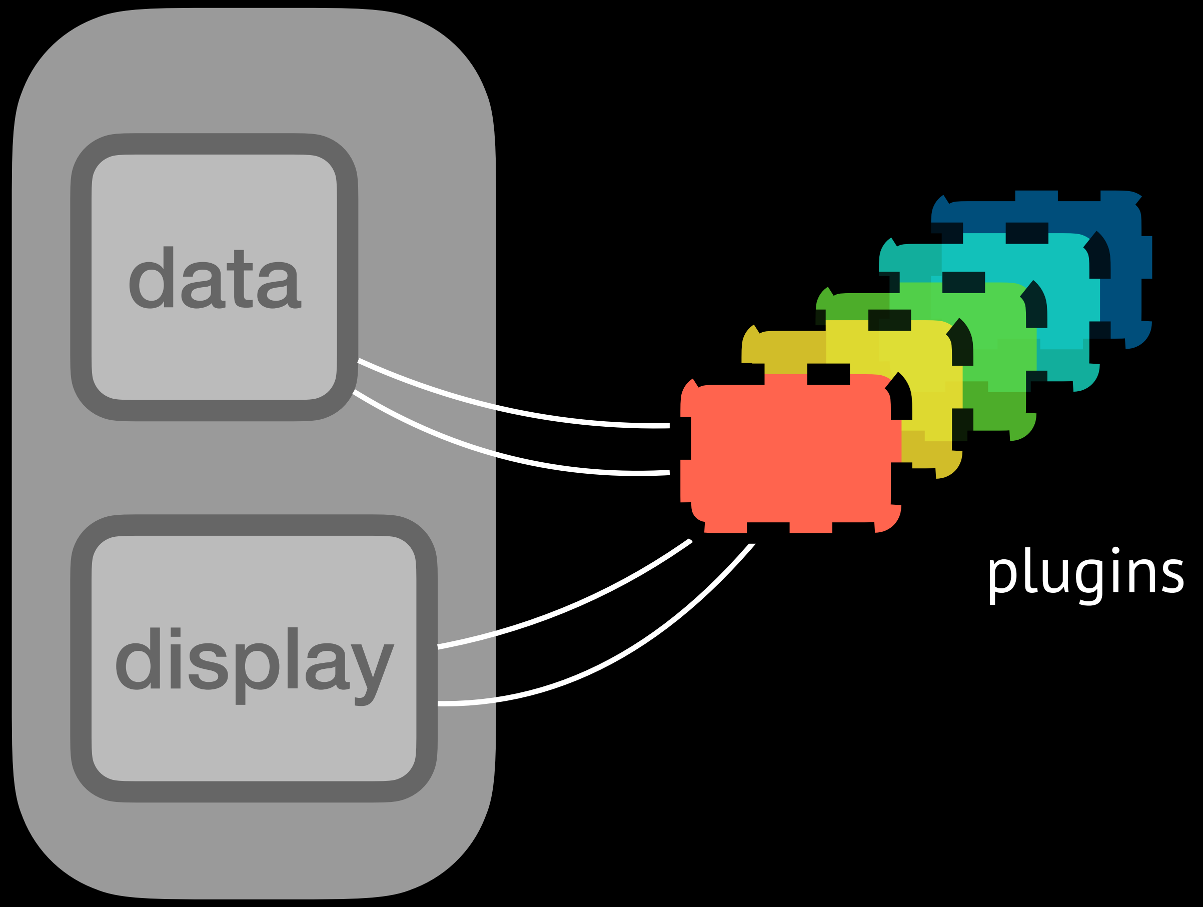
It is not enough that a system be accesible and easy to interface with, but it also must be allowed to grwo alongside the user. The user must be able to develop their own abstractions within the software, but also provide it with "sub-programs" that they write themselves or get from a third-party source. The concept of *plugins* allows a program to be extended in it's scope, providing arbitrary extra functionality in an integrated way.

In my view, it is up to a software designer to understand the dimensionality of a program, but not to define all the ways one can explore these dimensions. By allowing for additional design by users, a piece of software can be *drawn closer* to the individual, as a means of providing *custom fit*, the user can define their exact specifications.

Building this often involves the creation of a programming language, a language that is understood by the program that allows users to speak directly, or the development of a protocol language that outside code can conform to, an API that the extensible program extends to the extending programs to give them ability to speak to the hosting program.

> "an *extensional* system seemed to be called for in which the end-users would do most of the tailoring (and even some of the direct construction) of their tools."
>
> **Alan Kay**

> "There will never be enough professional programmers and system developers to develop or interface all the tools that users may need for their work. Therefore, it must be possible, with various levels of ease, for users to add or interface new tools, and extend the language to meet their needs."
>
> **Doug Engelbart**



Principle 9

Design

Semes and Spaces

Design Patterns

Minimize Visual Clutter

Allow Room to Grow

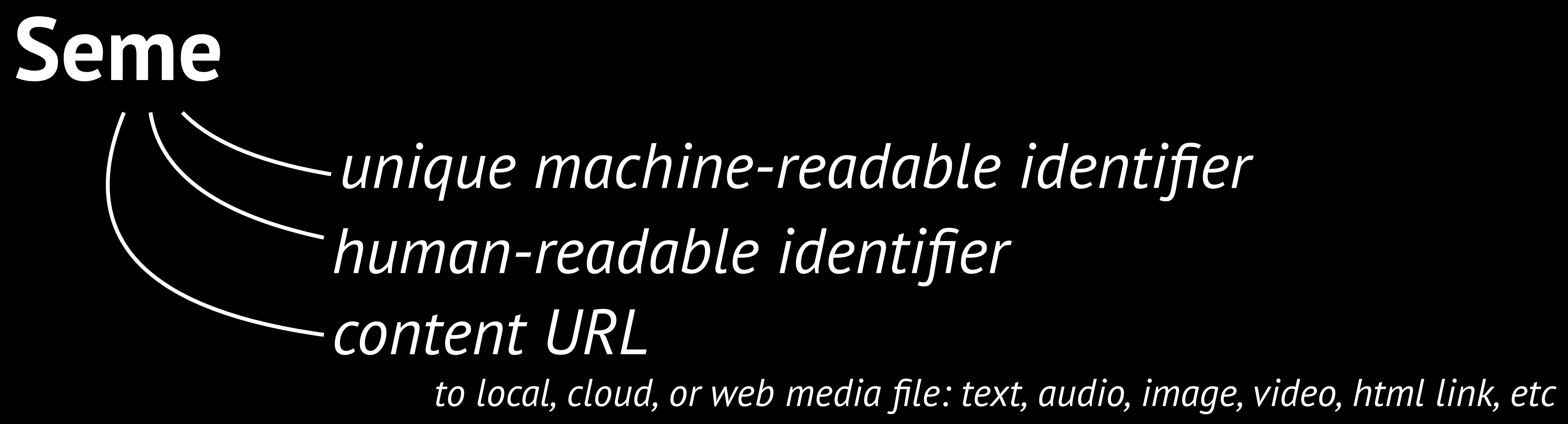
Balance linearity and non-linearity

Balance selection and collection

We can base our data model on a single primitive. Using the Object as Primitive principle, we create a unified interface to all of our concepts and information through one type: an Object. For our purposes, we'll refer to these objects as "Seme"s (defined by Merriam-Webster's as "any of the basic components of the meaning of a morpheme", or by Wikipedia as "the smallest unit of meaning recognized in semantics").

At the base level, at what we can consider to be the "leaf"-level of a tree, these objects hold a URL: a universal locator of information. This demonstrates our principle to "Play Well with the Outside World"—we use URLs to reach outside of our app's context, into local and external file-systems to find content. This is what allows Murlin to serve as a "super-structure" over a universe of content, the ability to refer globally to information, using file-system or Internet access to retrieve these into the app's context.

This also allows Murlin to play well with other apps, allowing powerful media editors, browsers and viewers to be utilized for their purposes, while being able to create structures that can refer to their addresses. These objects also need to be identified within the Murlin universe, both to human and machine. We can use unique identifiers like UUID to create names for Semes, and also give them human-readable names, through text and emoji.



Seme

unique machine-readable identifier

human-readable identifier

content URL

to local, cloud, or web media file: text, audio, image, video, html link, etc

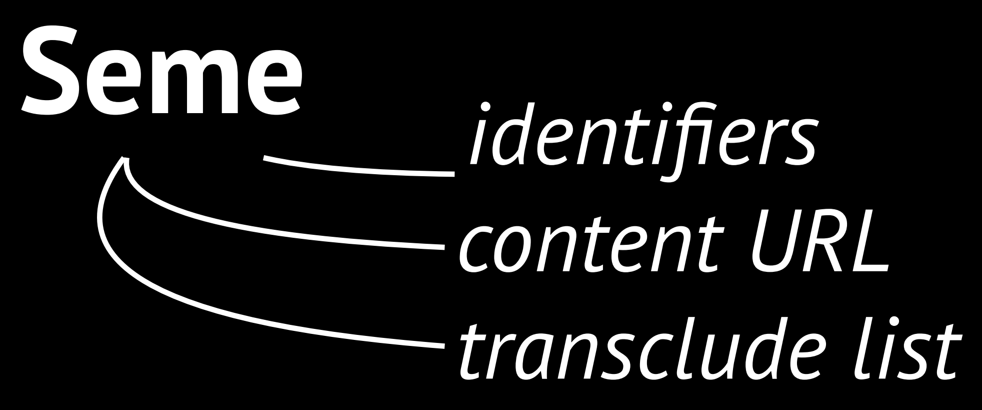
We can think of this layer of “leaves” like a key-value store, where we map ids to URLs, and provide means to fetch content in from the URL, or to visit the URL in it’s native context. We can begin to develop means to render and edit certain file types, and allow the creation, reading and writing of these.

We can accomplish this through the plugin pattern—an example of our Extensible principle. For media like text, graphics, spreadsheets, audio, PDFs and more, we can develop Murlin-native plugins to offer out-of-the-box solutions. We can also allow for other developers to implement their own set of media processing algorithms as plugins and allow these to integrate into the Murlin system.

This allows us to define a common set of file interactions and file format decisions, while handing developers raw data to process and return for display. We can control access to the visual system by offering an API to use to display controls, overlays and other *secondary* elements, while providing consistent foundational graphics.

It is not enough to just have a large dictionary of objects, we also need to express the interconnections between these objects. This is an expression of our principle of Link as Primitive. Instead of considering Objects and Links as separate, we can unify them into a single interface. We can simply grow our notion of what a Seme is.

Immediately, we can use the principle of Composition by Transclusion, allowing for the creation of Semes from selections, or the addition of selections to existing Semes. We'll call this action **Transclude.** We can accomplish this by simply adding a field and using this to track which the address and range of the Seme we are selecting from.



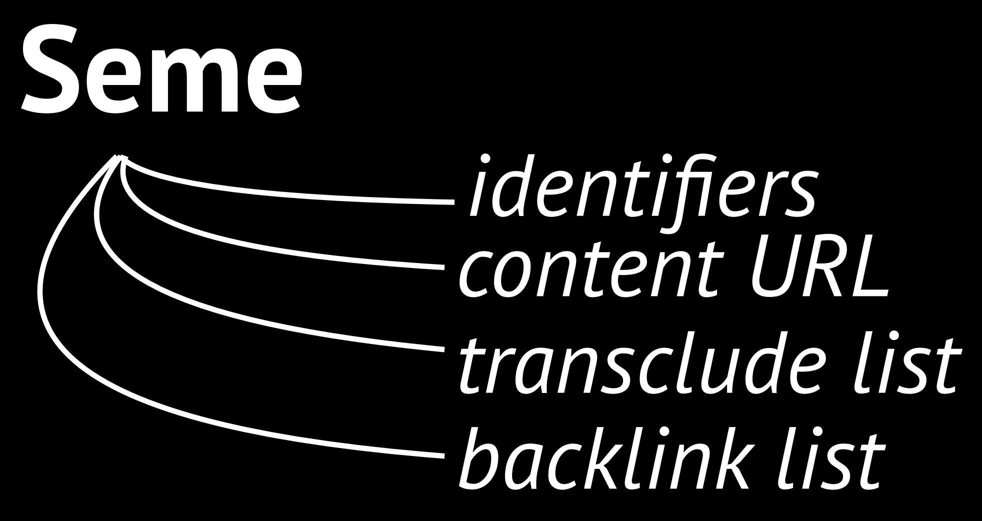
Seme

identifiers

content URL

transclude list

We can also consider tracking the backlinks—we'll want to track the references of which Semes have selected from our content. We can accomplish this as above, by adding another field that tracks backlinks.



Seme

identifiers

content URL

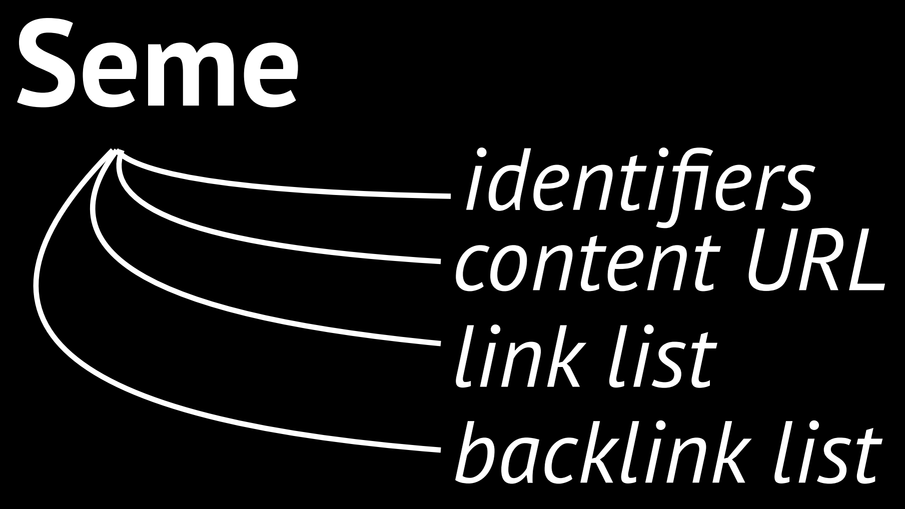
transclude list

backlink list

We want to balance our directions of growth. Just as we want to derive lower-order Semes, we'll want to integrate multiple Semes into a higher-order Seme. We can call this action Compose, enabling us to make compositions, collections and abstractions over other Semes.

This combination of f*usion* and f*ission* allow us to work along the abstraction axis, abstracting up and deriving down to create new Semes, or add to the collections of existing Semes. We think of moving laterally, creating copies of Semes. We can do this through the action Clone.

In order to integrate these new links into the current Seme data model, we can simply define a notion of a link, and continue to use the same fields. For Compose and Clone, we simply need a reference to the addresses of the linked Semes. For Transclude, we also need a range, to define the area of selection. This involves defining content-specific ranges, which understand the underlying media type. We can also store a timestamp with these links, to mark their creation.



Seme

identifiers

content URL

link list

backlink list

Seme

visual

displays ***rich media*** **or** ***rich space***

data

{

machine-id	id	1423e2-23a3…
human-id	id	“a made-up name”
content-URL	➔	file:///Users/a/made/up/path
links	⇔	[(machine-id, link-type, link-data)]
backlinks	⇔	[(machine-id, link-type)]
space-spec	➔	Internal File

}

operations

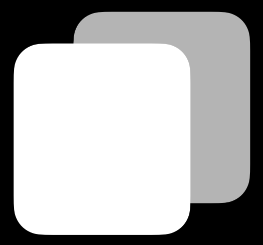
Compose

[■,■,■]

Transclude

■[:]

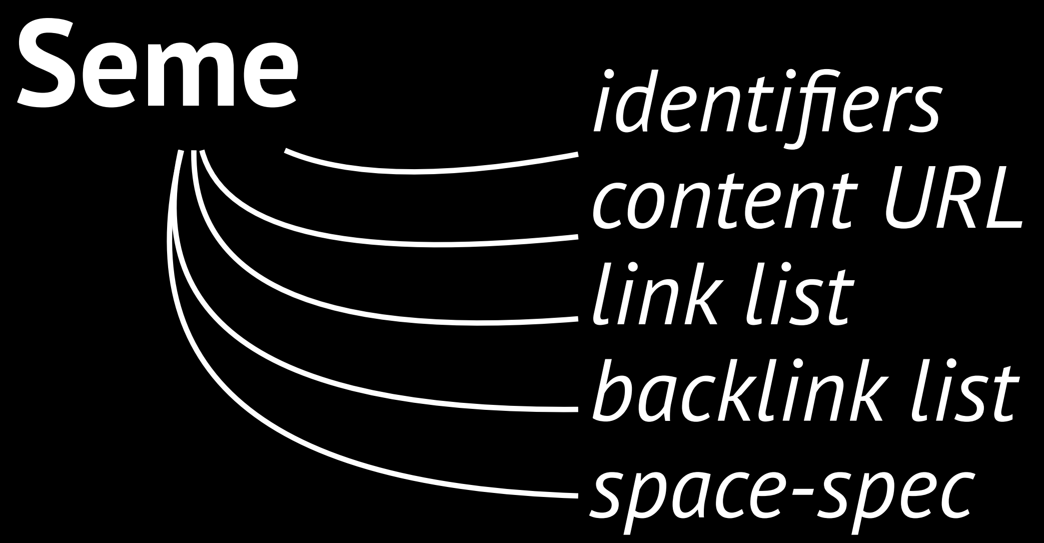
Clone



We now have a data material, comprised of a possible media url, and a possible set of connections to other units of the same material. We have ways of interacting with it, to grow more of it, in Transclude, Compose, and Clone. This allows the creation of a Personal Database, one of our principles, out of an interconnected universe of Semes—essentially, a graph database.

We now need to consider the graphic material to manipulate, and the spaces to manipulate it in. We turn now to our World Creation principle, that we should think of virtual worlds—not lists, documents, pages, walls, surfaces, boards, or canvases—*worlds*. This is not to say that we cannot use the structures of the above as elements, but that they are elements within a wider, grander space. We must not make the structure the space, rather we make a space for the structures.

In the same manner that we approached Transclude, Compose and Clone, we can approach Spaces and Structures—reuse the same data models. Instead of creating a new data type, we can simply use Semes as our core notion of Spaces: a collection of Semes that we wish to be present in our workspace, with attached data (in the form of a file) about the size of the space and the position and types of structures. This requires adding one extra data-field, for a file URL to our graphical spec, a JSON or XML-like file defining visual properties.

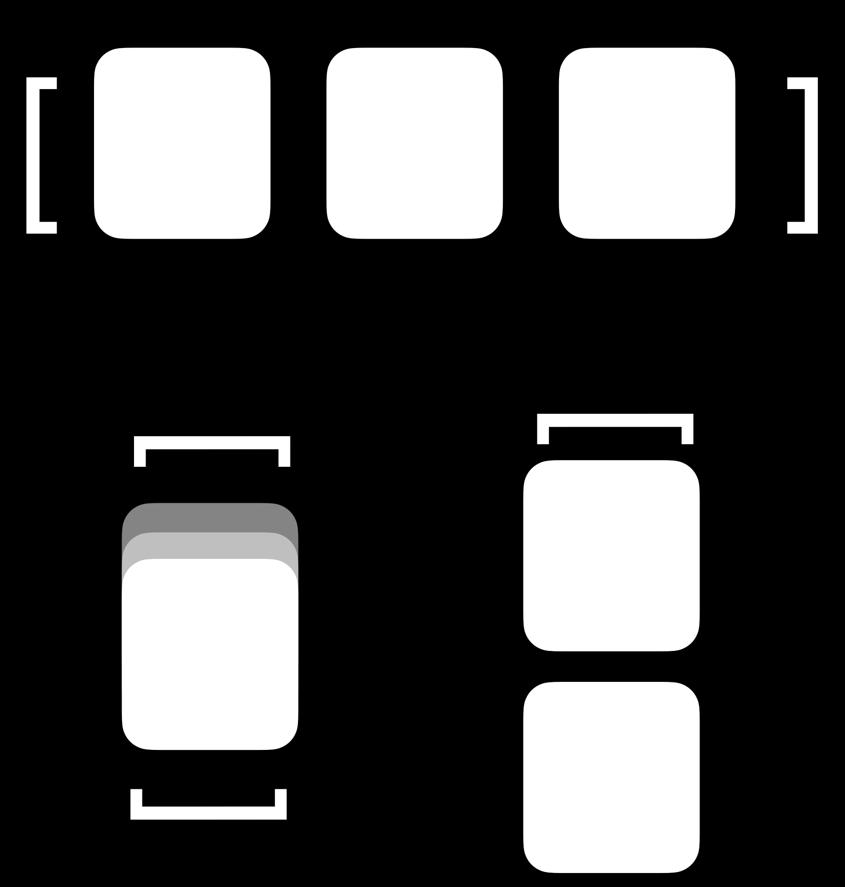


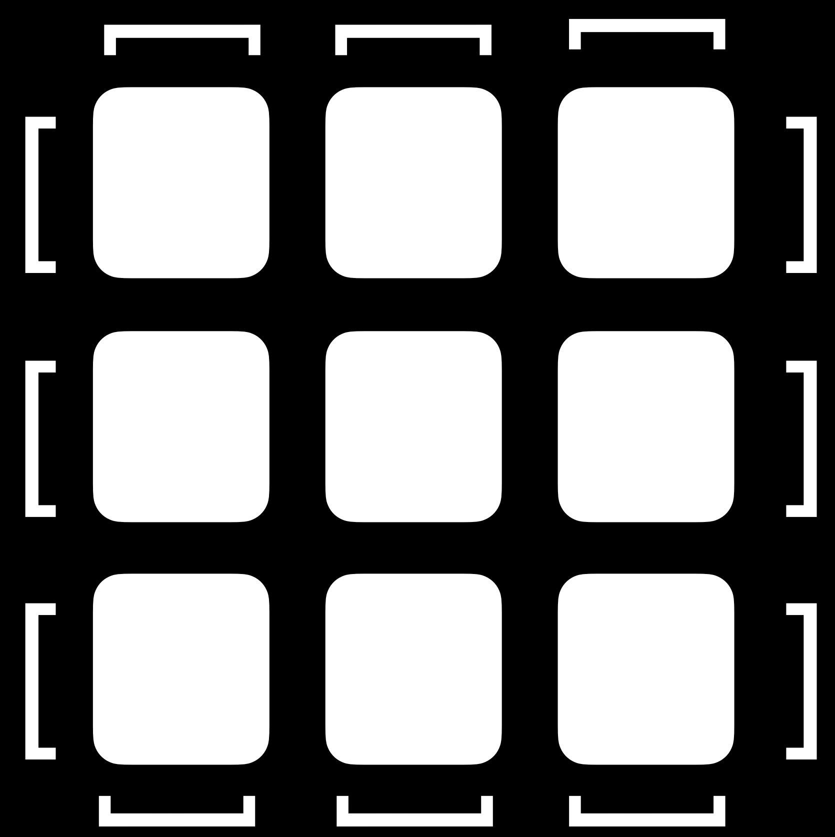
Seme

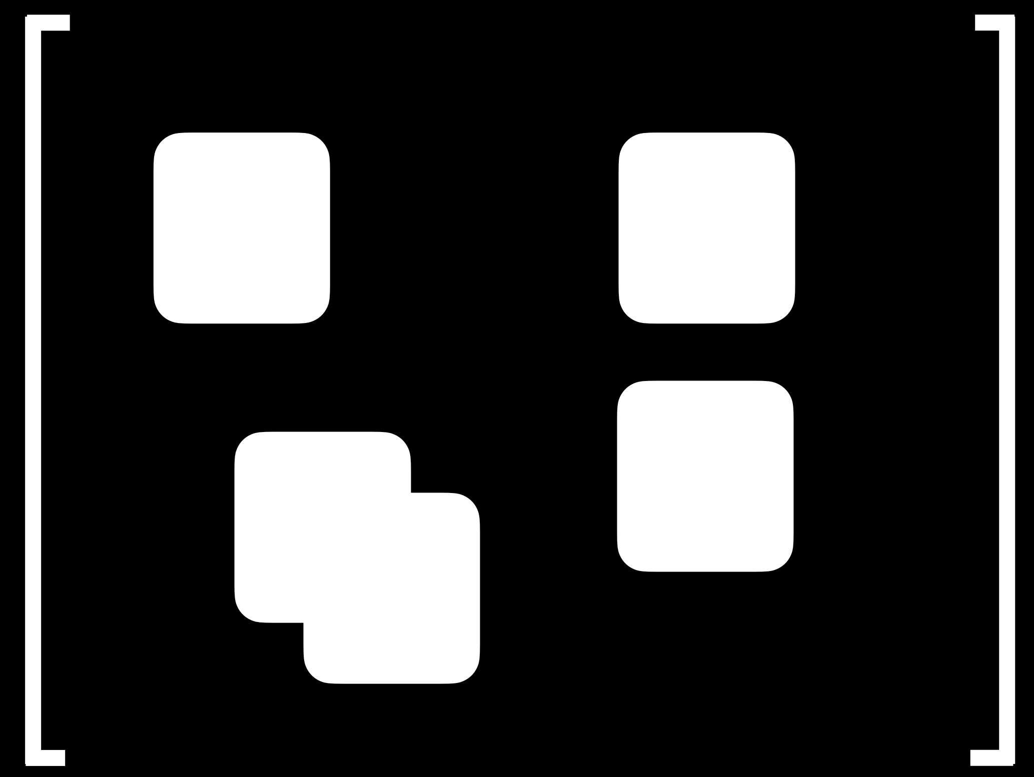
identifiers
content URL
link list
backlink list
space-spec

Space

"primitives"

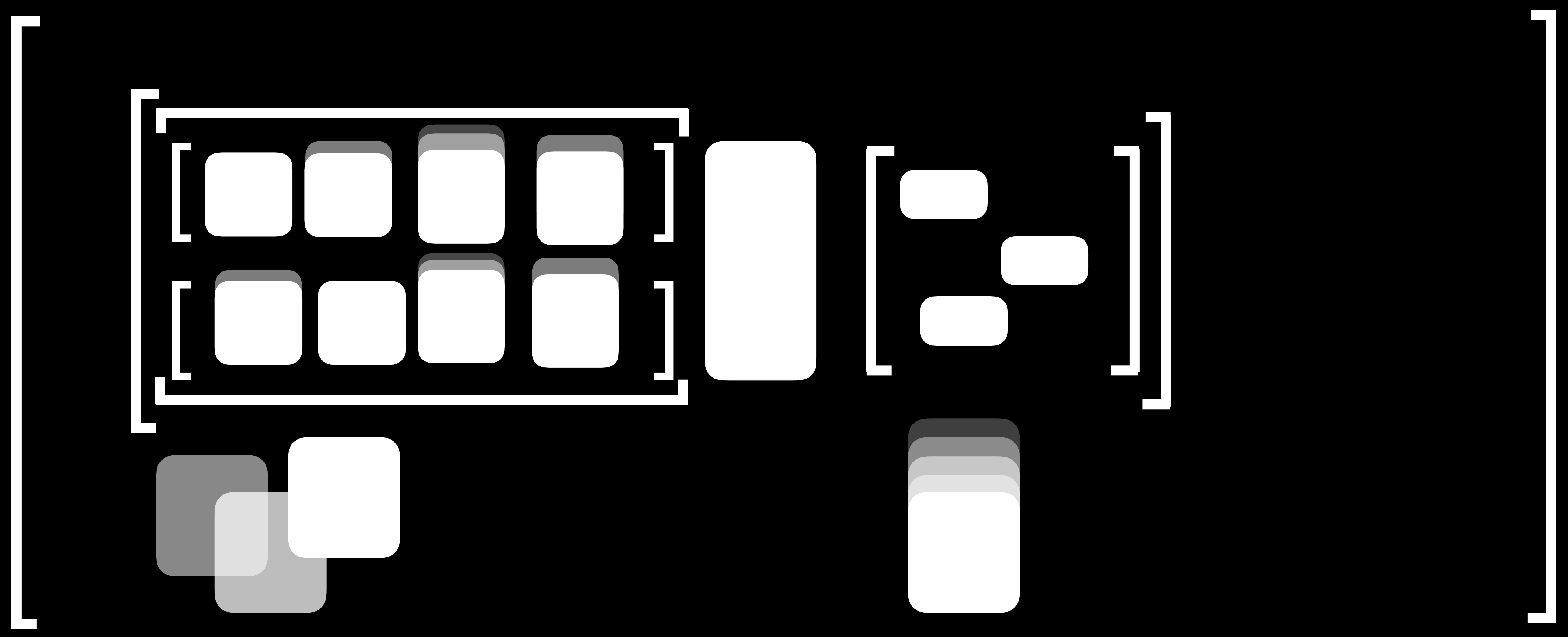






"compositions"

compose to make
more complex spaces



Taken together, these principles provide for a powerful, unified system in the spirit of Engelbart's "thought vectors in concept space", Bush's "associative trails", Kay's "personal dynamic media", Atkinson's "tool for making tools", Nelson's "two-way, visually connected documents" and others' dreams for a digital material suitable for embedding knowledge within.

These designs mark the up-to-date trajectory for **overlair's** Murlin project, describing the vision for the *digital material*. The next step is to figure out the *digital medium*—to figure out the environment in which Murlin will run; be it mobile, desktop, web, or game engine app. Currently, the platforms of iOS, React and Godot are being explored.

Currently, designs for Murlin are focused on a single-player application connecting flexible data models to flexible graphical representations. Looking forward, the goal is to be able to allow for multi-player access, as well as develop a way to interface with data and graphics programmatically.

Graphical and programmatic access to a personal, shared database