

Towards Agile Architecture

Objectives

This chapter will enable participants to:

- Identify the ingredients of an Agile Architectural posture of an organization
- Describe how the components come together to enable rapid but well-integrated enterprise systems



1.1 Pulling Together the Best of All of the Leading Practices

- Rather than taking a “magic bullet” approach let's take what works from the various disciplines and solutions:
 - ◇ Enterprise Architecture
 - ◇ Solution Architecture
 - ◇ Agile
 - ◇ Portfolio, Program and Project Management
 - ◇ Enterprise Packaged Software
 - ◇ Open Source
 - ◇ Cloud and SaaS
 - ◇ Continuous Integration

1.2 Enterprise Architecture [1/2]

- We know that:
 - ◇ Unmanaged duplication is a recipe for disaster

- ◇ Managed duplication is necessary for resiliency and sometimes performance
- ◇ Regardless of how many times the CFO asks for precise estimates they will be SWAG's
- ◇ Attempting to model an entire enterprise is a never-ending project that will not produce much if any value
- ◇ Packaged applications are a compromise: they give you capability faster than custom development but they rarely fit as well as expected
- ◇ Tools for polyglot integration such as Thrift, Avro and Protocol Buffers are becoming more popular
- ◇ Tools for Big Data are allowing us to capture data in raw, schema-less form

1.3 Enterprise Architecture [2/2]

- So:
 - ◇ We need some enterprise-level planning but just enough
 - ◇ We need clear enterprise-level standards, especially API's, to enable integration between business areas and applications
 - ◇ We need to bring Agile discipline to the process of developing architecture: we need to do Agile Architecture
 - ◇ Rather than trying to design a single logical data structure for the organization, just capture the raw data and provide access through standardized interfaces, allowing a schema to be applied on read
 - ◇ Focus on helping Agile teams to get data into and out of the store

Canada

821A Bloor Street West
Toronto, Ontario, M6G 1M1
1 866 206 4644
getinfo@webagesolutions.com

United States

436 York Road, Suite 1
Jenkintown, PA, 19046
1 877 517 6540
getinfo@webagesolutions.com

1.4 Solution Architecture

- We know that:
 - ◇ Every system has an architecture whether by design or default
 - ◇ Designing for a feature will save a huge amount over adding it later
 - ◇ It's impossible to predict all features that will be needed in advance
 - ◇ Adding hooks for features that aren't needed until later can slow down delivering what's needed now
- So:
 - ◇ Some Solution Architecture work could and should be done as part of an Agile project, or if cross-project following an Agile approach
 - ◇ It should still conform to the enterprise-level standards unless there is an economically justifiable reason not to

1.5 Agile

- We know that:
 - ◇ Custom solutions always fit best but aren't always cost-justified
 - ◇ A small dedicated team can move fast if the ceremony is stripped away
 - ◇ Building systems as isolated islands is not in the best interest of the enterprise as a whole
- So:
 - ◇ Agile is the way to go where custom development is truly called for

Canada

821A Bloor Street West
Toronto, Ontario, M6G 1M1
1 866 206 4644
getinfo@webagesolutions.com

United States

436 York Road, Suite 1
Jenkintown, PA, 19046
1 877 517 6540
getinfousa@webagesolutions.com

1.6 Portfolio, Program and Project Management

- We know that:
 - ◇ The tools of IT Portfolio Management provide an excellent framework for prioritizing the application of an enterprise's limited IT resources to the most valuable uses in the face of enterprises' seemingly insatiable demand for services
 - ◇ If done well it includes a broad swath of line management so there will be broad executive understanding of why the resources were focused the way they were
- So:
 - ◇ Draw on these techniques but keep the process short and focused, no more than 15 weeks
 - ◇ Don't attempt to model the enterprise any more than is necessary to identify the work, estimate it and prioritize it
 - ◇ Recognize that the estimates are literally order-of-magnitude and set the approval ROI accordingly high

1.7 Enterprise Packaged Software

- We know that:
 - ◇ For functions that are performed uniformly between enterprises following established standards and processes (e.g. accounting) packages are faster to implement
 - ◇ Packages are no easier to integrate than anything else
 - ◇ They're priced to be more attractive than custom development (but are still no bargain)

Canada

821A Bloor Street West
Toronto, Ontario, M6G 1M1
1 866 206 4644
getinfo@webagesolutions.com

United States

436 YorkRoad, Suite 1
Jenkintown, PA, 19046
1 877 517 6540
getinfousa@webagesolutions.com

- ◇ They come with their own architecture
- So:
 - ◇ Enterprise packages will remain an important part of the landscape and will dictate the architecture of the areas they cover

1.8 Open Source

- We know that:
 - ◇ Open source is continuing to grow in importance
 - ◇ The economics of open source make it compelling even for enterprise use
 - ◇ Employees and contractors will use open source components and tools whether you know about it or not
 - ◇ The open source community is nimbler and more responsive than corporate software vendors
- So:
 - ◇ Enterprises may as well embrace it and use it to get the resulting lift and agility

1.9 Cloud and SaaS

- We know that:
 - ◇ Cloud infrastructure is more flexible and agile than using dedicated servers
 - ◇ It can be less or more expensive, depending on the usage pattern
 - ◇ Services such as cloud git repositories, automated testing provide

Canada

821A Bloor Street West
Toronto, Ontario, M6G 1M1
1 866 206 4644
getinfo@webagesolutions.com

United States

436 York Road, Suite 1
Jenkintown, PA, 19046
1 877 517 6540
getinfousa@webagesolutions.com

- agility by avoiding needing to set up and maintain the infrastructure locally
- ◇ A wide variety of new SaaS solutions are continually coming to market
- ◇ SaaS is in principle less secure than locally-hosted alternatives but neither is truly secure unless extraordinary measures are taken
- So:
 - ◇ Cloud and SaaS will continue to increase in importance and need to be embraced
 - ◇ Standards and guidelines for their use are important and fit with the role of Enterprise Architecture

1.10 Continuous Integration

- We know that:
 - ◇ We can use well-defined API's to decouple architectures so they can co-exist rather than being all a single technology architecture
 - ◇ But, unfortunately API design hasn't settled into a standard form: we thought Web Services would give us a standard (but it crumbled under its own weight)
 - ◇ REST has promise but is not well-enough defined and causes data to be passed between systems whether required or not
 - ◇ Protocols like Thrift, Avro, Protocol Buffers and GraphQL show promise
 - ◇ Test-Driven Development is a powerful technique for developing reliable code but requires huge discipline
- So:

Canada

821A Bloor Street West
Toronto, Ontario, M6G 1M1
1 866 206 4644
getinfo@webagesolutions.com

United States

436 York Road, Suite 1
Jenkintown, PA, 19046
1 877 517 6540
getinfousa@webagesolutions.com

- ◇ Enterprise Architecture has a natural role in defining interfaces between systems built using Agile
- ◇ Use rigorously specified, well-designed API's with continually updated documentation to allow Agile teams to work independently but still integrate well
- ◇ EA should enforce use of Test-Driven Development by Agile teams

1.11 So How Does This All Come Together?

- The team needs to think and work in terms of sprints: weekly deliverables rather than an annual or other cycle
- The focus needs to be on the area of greatest immediate need/value for the enterprise
- The team must work to fill in the gaps between Agile teams in addition to working on those teams
- Minimize ceremony, tools overhead, modeling for modeling's sake

1.12 Modeling

- Wasn't that a fundamental tenet of EA?
- Actually, none of the major EA frameworks emphasize it (although many general EA courses do)
- We haven't converged to an agreed modeling approach: (Accountants did it, why can't we?)
- Process and entity models are at about the right level of detail for EA work but don't fit well with an object-oriented development process
- Use Cases have stood the test of time, however UML generally hasn't

Canada

821A Bloor Street West
Toronto, Ontario, M6G 1M1
1 866 206 4644
getinfo@webagesolutions.com

United States

436 York Road, Suite 1
Jenkintown, PA, 19046
1 877 517 6540
getinfousa@webagesolutions.com

- ◇ Very few understand it well
- ◇ It's too fine-grained for EA
- ◇ It is of limited use for package implementation projects
- Functional programming languages (which don't follow an Object-Oriented paradigm like UML does) are gaining in popularity, introducing yet another “impedance mismatch”
- So: Do the minimum necessary, perhaps not at all

1.13 Reference Architecture

- Refocus from an attempt to impose a single Reference Architecture on the entire enterprise to defining boundaries and interfaces
- Common infrastructure services are still preferable but in practice very few enterprises have achieved it (and come the next merger they'll be middle-of-the-pack again)
- Industry-specific Reference Architectures will continue to be important and will need to be tracked and conformed to

1.14 Organization and Governance

- A centralized group is still required to take an enterprise view and define the inter-project boundaries and interfaces
- The team members need to adopt an Agile rather than slow consensus mindset
- The Agile Architecture group needs to spend as much or more time with the Agile development teams as with the other EA's

Canada

821A Bloor Street West
Toronto, Ontario, M6G 1M1
1 866 206 4644
getinfo@webagesolutions.com

United States

436 York Road, Suite 1
Jenkintown, PA, 19046
1 877 517 6540
getinfousa@webagesolutions.com

1.15 Requirements Gathering

- Decomposition should proceed incrementally
- No “boil the ocean” projects (especially planning projects)
- Support or participate in rapid prototyping to validate architectural requirements
- Always ask how much supposed requirements are worth i.e. how much would you be willing to pay for this?
- Be ruthless in differentiating “wants” from “needs”

1.16 Requirements Management

- Actively feed and consolidate requirements from Agile development teams
- Actively validate and re-validate priorities with senior management
- Spend no time on tools issues—use pen and paper if necessary

1.17 Portfolio Management

- Project prioritization is still every bit as important
- Do annual 13 week max. time-boxed strategy reviews/updates in parallel with the more focused sprints
- Include a wide swath of senior management so they understand each others' priorities and why some areas are getting greater attention than others

Canada

821A Bloor Street West
Toronto, Ontario, M6G 1M1
1 866 206 4644
getinfo@webagesolutions.com

United States

436 York Road, Suite 1
Jenkintown, PA, 19046
1 877 517 6540
getinfousa@webagesolutions.com

1.18 Deliverables

- Pare the list of formal deliverables to the minimum
- If you use an architecture development method question the value of each deliverable it identifies
- Ask the Agile development teams what they need to clear their paths

1.19 Architectural Compliance

- Insist that interfaces be designed in an application-neutral and flexible manner but spend less time on the architecture of the projects themselves
- Encourage quality through Test-Driven Development and peer reviews rather than formal Boards

1.20 Education

- Add training in Agile to the agenda
- Make sure the team understands integration technologies: ESB, Thrift, Kafka, XML, JSON, GraphQL, etc.
- Study the formal architecture methods but then dissect them and adopt what makes sense—dispense with the rest

1.21 Stakeholder Management and Communication

- Adopt the Agile perspective: think of stakeholders as partners or team members, not outsiders to be managed
- Continually communicate within the Architecture team, to the stakeholders and to the Agile development teams—this is a form of integration in and of

Canada

821A Bloor Street West
Toronto, Ontario, M6G 1M1
1 866 206 4644
getinfo@webagesolutions.com

United States

436 York Road, Suite 1
Jenkintown, PA, 19046
1 877 517 6540
getinfousa@webagesolutions.com

itself

1.22 Summary

- Architecture can be Agile!
- Use sprints to rapidly develop valuable, focused, incremental architectures
- Drive out ceremony and procedure
- Drive out wasted time on tools arguments
- Focus on the interfaces between systems and not the systems themselves
- Support an environment for continuous delivery
- Do just enough enterprise-wide analysis to be sure the Agile teams are focused on the right problems

Canada

821A Bloor Street West
Toronto, Ontario, M6G 1M1
1 866 206 4644
getinfo@webagesolutions.com

United States

436 YorkRoad, Suite 1
Jenkintown, PA, 19046
1 877 517 6540
getinfousa@webagesolutions.com