

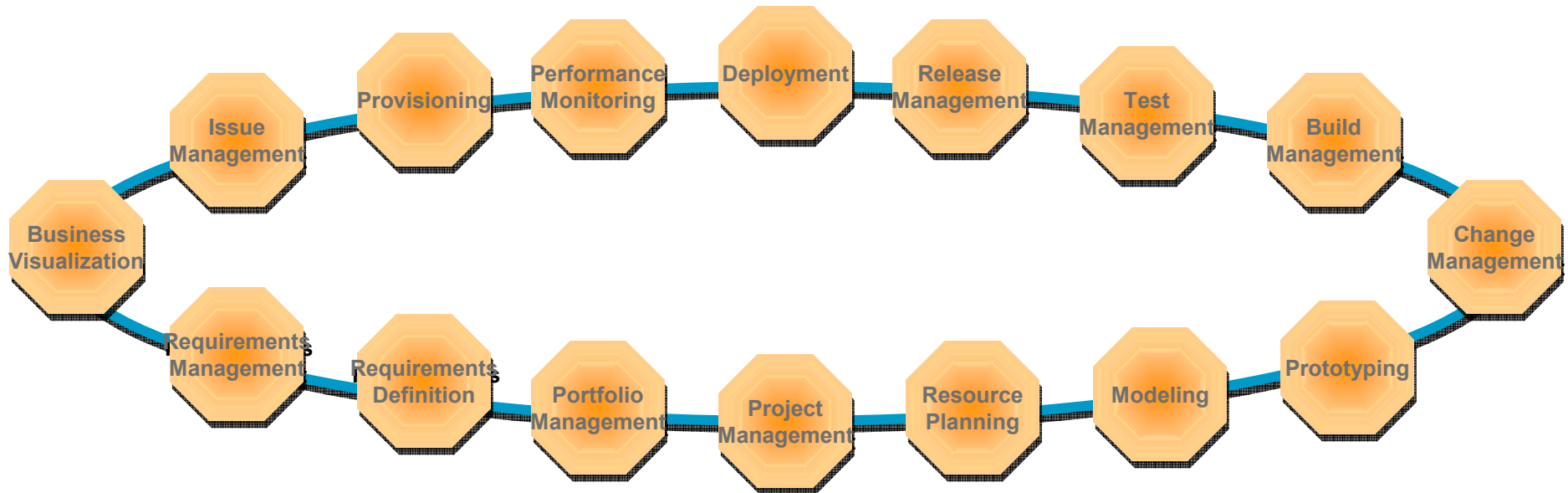


# SERENA<sup>®</sup> Eclipse Member Presentation

Brian Carroll, ALF Project Lead  
January 23<sup>rd</sup>, 2007



# Dream: What the customer thinks they bought

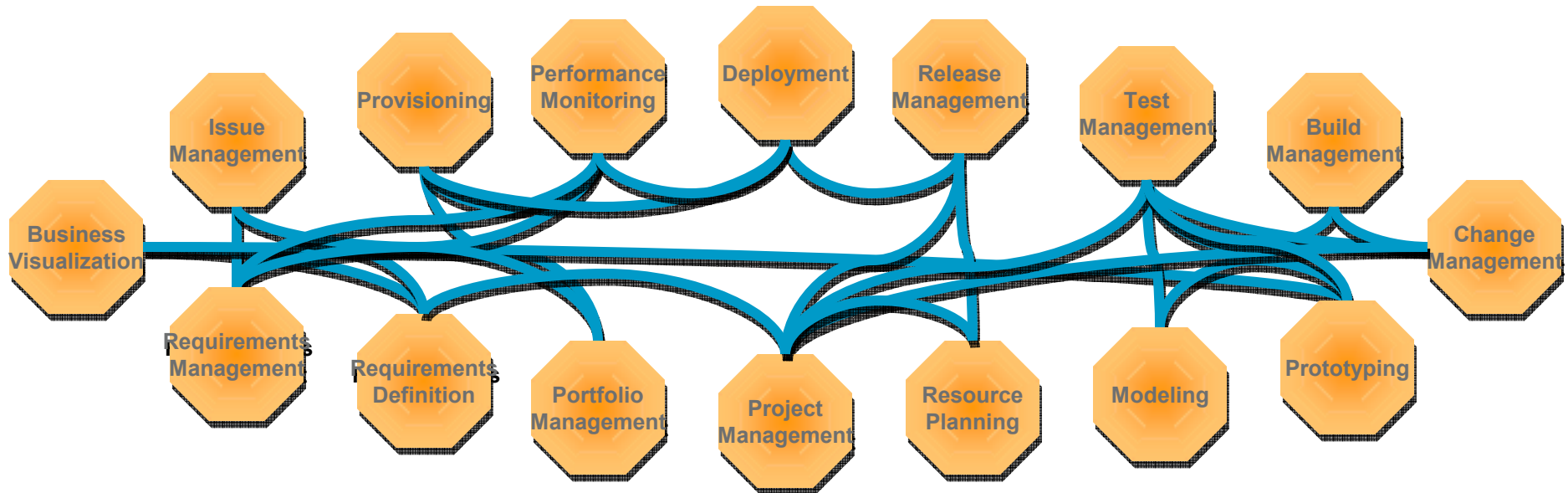


Customer view:  
Nice in theory but a heck of a lot of work

Vendor view:  
If only it were that simple



# Nightmare: What the customer is implementing



Customer view:  
Frail, fragile and, frankly, feeble

Vendor view:  
We can't keep up with the rate of change





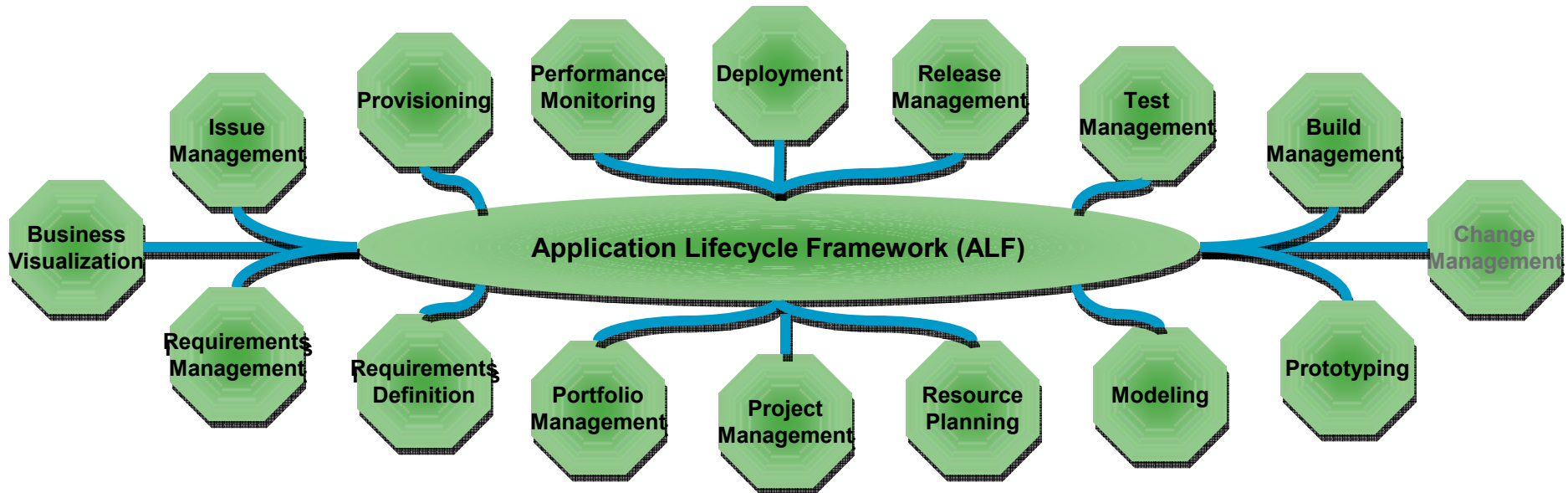
# Why is this a problem?

- Cost
  - Point-to-point integrations are expensive to write and maintain
  
- Quality
  - Point-to-point integrations are difficult to keep current
  - Point-to-point integrations are costly to test
    - Many product release combinations
  
- Risk
  - Should you bet your development on fragile integrations?



# Time to wake up:

What the customer wants – a standard tool “wall plug”



Customer view:  
It's about time

Vendor view:  
The only way we grow the market



# Why not just integrate on the Eclipse desktop?



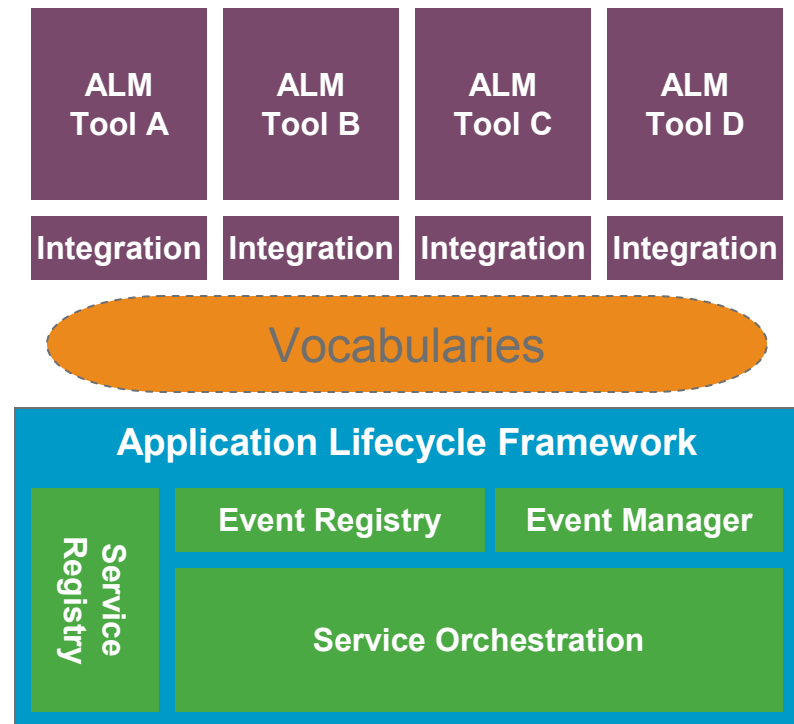
- Sadly, not all tools are Eclipse-enabled
  - Some are browser only
  - Some run on the mainframe
  
- Variety of interfaces exist
  - Command line, API's, web services, Eclipse UIs, native desktop UIs, web browser UIs
  - Can eliminate cut and paste
  
- Some tool synchronizations involve lengthy data transfer
  
- ALF focus is on integration at the server



# Introducing the Application Lifecycle Framework (ALF)

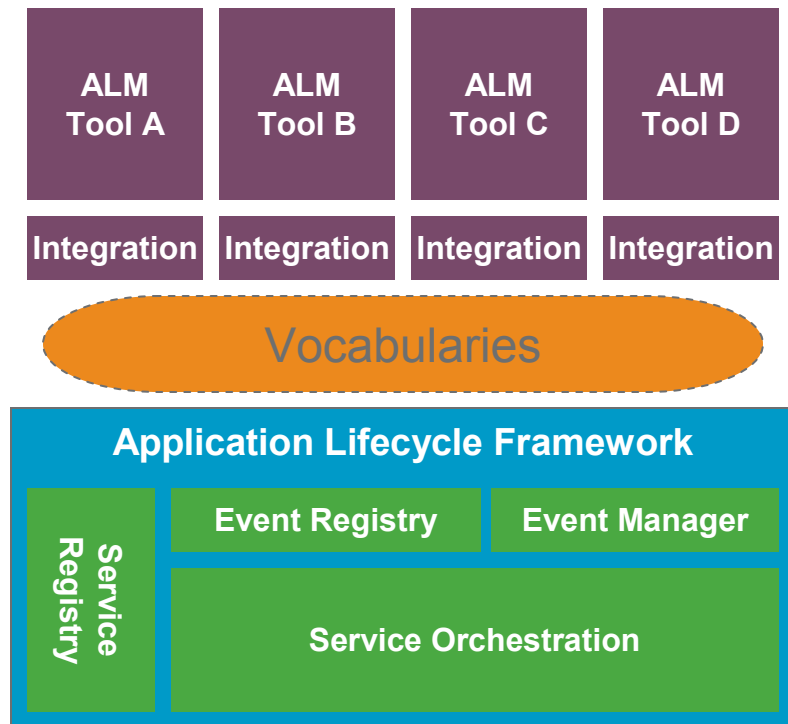


- A common platform
- Eclipsed based architecture
- Support for heterogeneous platforms
- Vendor neutral
- Standards based
- Extensible Framework





# What are the components?



- Event Manager
  - Event filtering and routing
- Pluggable BPEL engine
- Vocabularies
  - Initially SCM and BUILD
- Administration tool
- Identity and Single SignOn
- Guidance
- Conformance program





# Consumable components/artifacts from ALF

- Event Manager - Event router and filter
- Single SignOn and conveying identity over WS\*
  - Security token format – SAML based using claims
  - SSO Servlet Filter for web resources/web services
- Utility web services
  - Logging Service
  - Notification Service
- Guidance and best practices
  - Conformance guidance
    - From “come as you are” to full conformance
  - Best practices for WS and BPEL
  - Event definition
  - SAML-based security token format



# What ALF consumes today



- Higgins Security Token Service – Security token issuing and validation
- Currently using ActiveEndpoints for proof of concept



# What ALF would rather consume than build

- BPEL editor, or a BPMN editor with a mapping to BPEL
- BPEL execution environment
- Data mapping editor and a BPEL-callable mapping service
- Relationship Service – Cross product relationship store
- Human Interaction interface
- JAAS authentication hooks for plug-ins – Jazz contribution to Equinox?
- Rule-based way to extend WTP validation engine
- ECF provider for web services



# ALF Status



- Preparing for first release review - 0.7 release
  - Events and orchestration
- Adding single sign-on and identity in next release 0.8
- Preparing the software and developing the community to exit incubation
  - Active developer and extender community
  - Working to develop the end user community





## Suggested Next Steps

- Link Corona events to ALF Events
  - Use ECF
- Link Kepler team workflow with ALF process automation
  - Kepler emits event to trigger ALF process
  - Kepler human interaction is web-service enabled
- In preparation for EclipseCon, create a poster/graphic or whitepaper to explain how the projects fit together
- Coordinate with STP on BPEL editors and environment





**Thank you!**

