Converge: A Framework for Rapid Development of Converged Telecom Applications

Gregory Bond
Innovative Services Research Dept.
2009-12-10
what is a telecom app and how does it affect a call?

- telecom sessions can be affected by applications at session establishment time or after session has been established
- telecom sessions persist
• what is a telecom app and how does it affect a call?
• telecom sessions can be affected by applications at session establishment time or after session has been established
• telecom sessions persist
Effecting Control in an Isolated Telecom World

- A telecom app isolated from the external environment can only react to telecom-oriented signals: audio (video) or DTMF.
Goal: Ending the Isolation

- what does opening a telecom app to its environment offer?
- ability to share DB–based data – DB access is inherently **indirect** means of communicating with environment
- telecom app should be able to **directly** update environment of its status and vice versa (e.g. for control) – don’t want to poll shared data for updates
- want to accomplish this goal in a standardized way so that apps can operate on different vendor equipment
Goal: Ending the Isolation

- what does opening a telecom app to its environment offer?
- ability to share DB-based data – DB access is inherently **indirect** means of communicating with environment
- telecom app should be able to **directly** update environment of its status and vice versa (e.g. for control) – don’t want to poll shared data for updates
- want to accomplish this goal in a standardized way so that apps can operate on different vendor equipment
Goal: Ending the Isolation

- what does opening a telecom app to its environment offer?
- ability to share DB-based data – DB access is inherently **indirect** means of communicating with environment
- telecom app should be able to **directly** update environment of its status and vice versa (e.g. for control) – don’t want to poll shared data for updates
- want to accomplish this goal in a standardized way so that apps can operate on different vendor equipment
Solution: Merge Telecom and the Web

- two existing, similar technologies: SIP servlets for telecom and HTTP servlets for the web
- merging them means that telecom has access to the web and vice versa
Result: JSR 289

- JSR 289 standard finalized August 2008
- A container (e.g. jboss, tomcat, sailfin) runs servlets (server side programs) and can provide local services to servlets e.g. database connection pooling
- Apps can have sip and http components that are able to interoperate
Why Cheer for a Collision?

- Services are moving to a web-based cloud - now possible to move telecom to the cloud
- Novel apps possible when telecom integrated with the web e.g. telecom mashups
- Telecom app development can benefit from mature web app development environment
Next Problem: Isolated Development Frameworks

- it’s not enough to have merged HTTP and SIP servlets
- no one in the web world programs with ‘raw’ HTTP servlets – web developers use frameworks
Digression:
What’s a Framework?

• Runtime libraries and development tools that support an abstract view of the application

• Some web app frameworks: Spring, Struts, ASP.NET, Django, Ruby on Rails, Grails

• Only one telecom app framework: our own E4SS (ECharts for SIP Servlets)
The Sequel

SIP Servlet Framework

HTTP Servlet Framework

!?
The Converge Meta Framework

\[ \text{Converge} = \text{E4SS} + \text{Grails} \]

- Why Grails? generates HTTP servlets and closest to Ruby on Rails level of abstraction.
Grails is an open source, “full stack” framework including an ORM (Object Relational Mapping) DSL for Hibernate (supporting DB persistence), web page templating via SiteMesh, session/transaction management via Spring, MVC architecture (model–view–controller), convention over configuration (CoC)
a reusable E4SS telecom component provides two interfaces: S2J for accessing/notifying the external environment from the component, and J2S for accessing/notifying the component from the environment

developer simply needs to customize the S2J interface to suit their app and call the J2S interface when appropriate
Converge includes libraries that permit a Grails app to access an E4SS app and vice versa while co-existing in the same servlet.
• AttendedTransfer requires FeatureAddress data to store Expert’s number and FeatureInstance data to store current call state (since feature operation is conditional on call state)
• web site used to check call status and effect call control e.g. consult with expert
Example: Reusable Attended Transfer

- J2S methods effect control – consult() includes consultant address
- S2J methods provide notification
• AttendedTransfer requires FeatureAddress data to store Expert’s number and FeatureInstance data to store current call state (since feature operation is conditional on call state)
• these pre-defined domain classes are much of what are included in the ConvergeCore plugin
• we’ve encountered these domain classes many times over a decade of telecom development so Converge makes them available for re-use now
• feature data classes are abstract and intended to be subclassed by developer
• FeatureData common across all instances of a feature regardless of subscriber e.g. a media resource address
• FeatureAddress data is subscriber-specific data e.g. time of day for do-no-disturb
• FeatureInstance data is call-specific data e.g. current call state
Attended Transfer
Browser Interface

- this took me a couple of hours to write
- Converge automatically generates admin web pages and skeleton user page
- ‘Consult’ button is evidence of FeatureInstance data in play here
Converge Development Kit

Converge Runtime
Converge Runtime

- can run any sip servlet app
- includes 3 most common apps in order to get started fast
- powerful DFC application router supporting app composition
- database backed persistent provisioning and subscriber data
- web-based admin
Converge Runtime

- can run any sip servlet app
- includes 3 most common apps in order to get started fast
- powerful DFC application router supporting app composition
- database backed persistent provisioning and subscriber data
- web-based admin
- can run any sip servlet app
- includes 3 most common apps in order to get started fast
- powerful DFC application router supporting app composition
- database backed persistent provisioning and subscriber data
- web-based admin
Converge Runtime

- can run any sip servlet app
- includes 3 most common apps in order to get started fast
- powerful DFC application router supporting app composition
- database backed persistent provisioning and subscriber data
- web-based admin
• home page of Converge runtime web-based admin
take a step back now and consider contexts where Converge can be used
converged apps can be deployed in one of these four ways
Frameworks vs Web Services

• Different access points to telecom logic:
  • Web services - more protected access to network, suitable for 3rd party (e.g. small business) and client-side mashups
  • Framework - wider access, suitable for service providers and trusted 3rd party

• Web service performance/latency issues

• Frameworks can help expose web services
Converge Development Models

- Develop web services for telecom apps
- “Drop in” reusable telecom component for a custom converged app
- Concurrently develop custom telecom component and web app
Summary

• Converge released as open source on echarts.org at JavaOne 2009
• currently runs on Sun/Oracle SailFin
• porting to RedHat Mobicents Tomcat/JBoss

• Two other converged development frameworks announced at JavaOne 2009:
  • RedHat JBoss Seam Telco
  • Sun/Ericsson SailFin CAFE