



Java is a trademark of Sun Microsystems, Inc.



JavaOneSM

Castle in the clouds - SaaS Enabling JavaServerTM Faces applications

Lucas Jellema

AMIS, The Netherlands

<http://technology.amis.nl/blog>

Overview

- > What makes an application 'SaaS'?
 - And why provide an application as SaaS
- > Requirements of SaaS applications
- > Focus on Customization
- > Integration & SOA across the cloud
 - Deep Linking, Mash Up and UI Services
 - Services API
 - Remote Events Listeners
- > Summary and Road Ahead



Software as a Service

common characteristics

- > Single instance of standard application & database
- > Running in a shared infrastructure, 24/7
- > Managed by central administration team
- > Delivered over the internet to web browsers
- > Used by individuals from multiple organizations
- > globally distributed & multi-lingual user community
- > Licence model based on usage metrics
- > Service Level Agreement to specify quality of service



Software as a Service (perceived) benefits

- > Small up-front investment
- > Quick startup – ‘turn on the software tap’
- > Professional administration with guaranteed performance and application availability
- > Low TCO through ‘economies of scale’
- > Dynamic Scalability
- > Rapid evolution of new features and bug resolution – quick deployment
- > Quick adoption of new technology
- > Community to foster and share best practices

Software as a Service concerns & potential disadvantages

- > Dependence on 3rd party for
 - Security, Availability & Performance, Scalability
- > Data outside the own infrastructure
- > Standard application – not at all tailor-made
 - Our own terminology, fields, default values, validation rules, workflow
- > How does SaaS application integrate with
 - Our SOA & EDA, Mash Up/Portal and Workflow
 - Identity Management and Single Sign On
- > How to prevent lock in with the SaaS provider

SaaS Requirements

- > Integration across the cloud
 - Make SaaS application integrate with the SOA & EDA, workflow management, identity management and other shared central facilities in the organization using the SaaS
 - Two way, based on open standards, geared towards both programmatic and UI integration
- > Customization
 - Allow organizations and individuals to tailor the 'standard' application for better fit
 - Within the single instance of the SaaS application
 - Easy, dynamic, run-time, self-service in browser

Application 'levels of reality' Factory

Customer



Organization A

End User



Customer



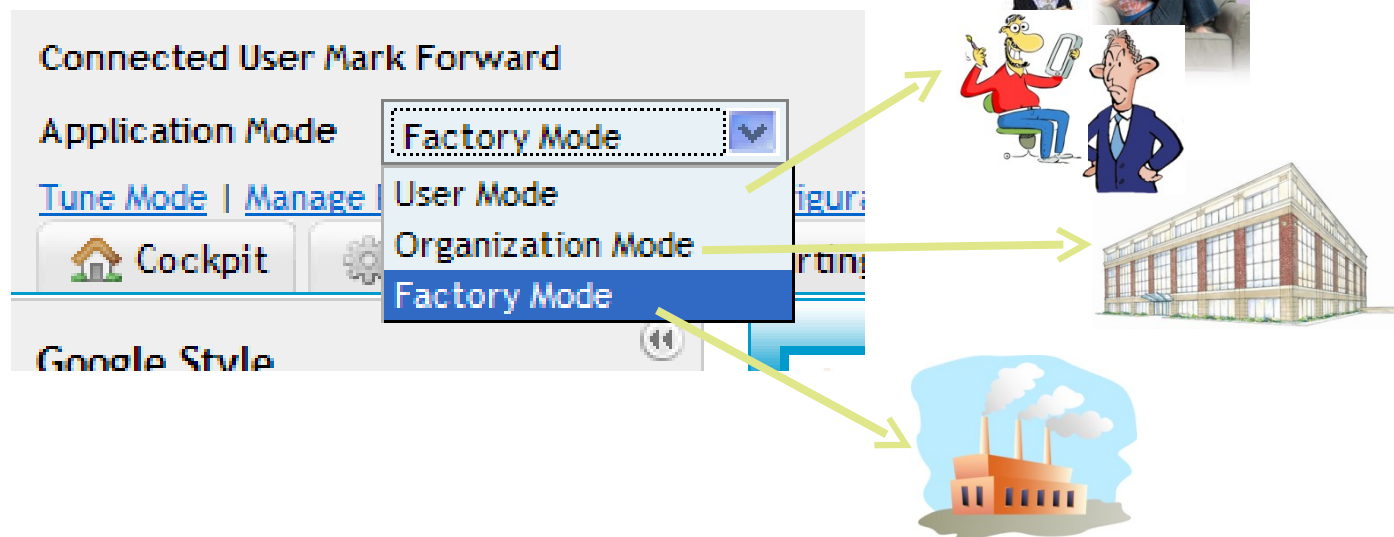
Organization B

End User



Application Mode Switcher

- > Application Managers can switch between Application Modes
 - To view and manipulate the application at the selected level



Application Rollout process...



- > Development in the Factory
- > Production-time refining



- > Customer (organization) specific implementation
 - By SaaS Helpdesk, on site consultant, Application Admin



- > End User (dynamic, run-time) Personalization

Proposal

- > Let's approach every end user application as a SaaS application!
 - Both standard and custom application

Desirable SaaS features & facilities

- > Dynamic, Wiki like, context sensitive help
- > Integration with incident registration
 - Automatic passing of page, context data & logging
- > ‘Simulate user’ option for the service desk
- > SLA metrics and application usage monitoring
- > Self Service style application management
- > ‘Toggle’ between factory, organization and user
- > Integrated workflow & collaboration
 - Communication (email, chat), task list, tagging, ...
- > “Register interest in” – notification of events

Customization

- > Allow adaption of terminology to organization
 - Resource Bundles per organization (and individual)
 - Dynamic, run-time resource bundle management
- > Support for adding/hiding fields
- > Configuration of application behavior through organization and user level preferences
- > Personalization (my...) features for fine tuning
- > Workflow and task-scope configuration
- > Configure menus and add new items
- > Fine tune look & feel – e.g. skins, styles, logos

Customizing terminology



- > Boilerplate text – prompt, title, button label, messages, help,... - come from resource bundles
 - Built in support for i18n – language & country
 - Typically held in property files deployed in WAR
- > We need to implement resource bundles
 - that can be updated and refreshed at run time
 - Or at least extended with any overrides of default values
 - keys that support the notion of organization specific versions of bundle entries in addition to factory ones
 - maintenance facilities with self service support
 - That presents bundle entries in understandable format

Implementing Self Service Customization of Resource Bundles

- > Resource Bundle implemented as class (extend ListResourceBundle) that read from database
- > Keys are constructed as: <base_key>_<org_id>
 - And even <base_key>_<org_id>_<user_id>
- > JSF page refers to resources through intermediate bean: #{bundleHandler['baseKey']}
- > The bundleHandler bean first tries user specific key; if not found then org key and then base key
 - It can also access several resource bundles – first the dynamic, database based one with overrides

Custom(izing) fields

- > Some fields are never used by some organizations and are therefore ‘in the way’
 - or they may even be legally forbidden in a country!
- > Every organization finds the application lacking with one or more fields
 - but no two organizations agree on which fields...
- > To allow optimal customization, a SaaS application ideally allows
 - Customization of standard items
 - Addition of custom items embedded in pages
 - And associated with data records like ‘regular’ items

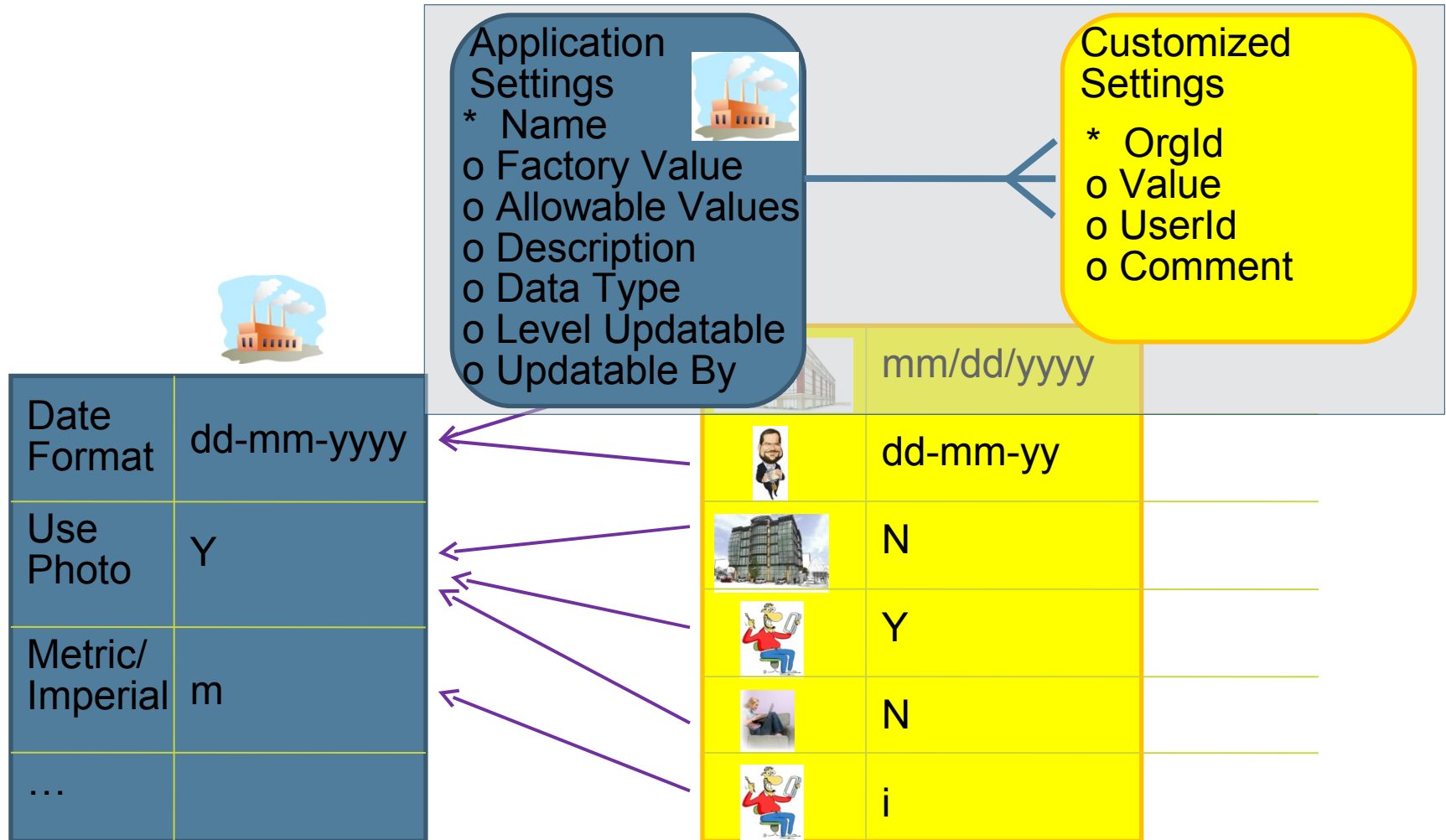
Application Settings for customizing application behavior

- > Some Application Behavior is developed as *dynamically configurable*
 - Through Preferences or Application Settings
 - *Display*: Locale, Format, Units for Height and Weight (converter parameters), feature on/off
 - *Business logic*: Default Values for new records, Validation parameters, Field & Button Authorization
 - *Styling*: Images, Fonts, Colors, Skins/Themes
- > Settings can be configured at Factory (default), Organization and User level
 - At run-time through self service pages

Integrating Application Settings during development

`#user.settings['NAME']`

Application Settings Data Model

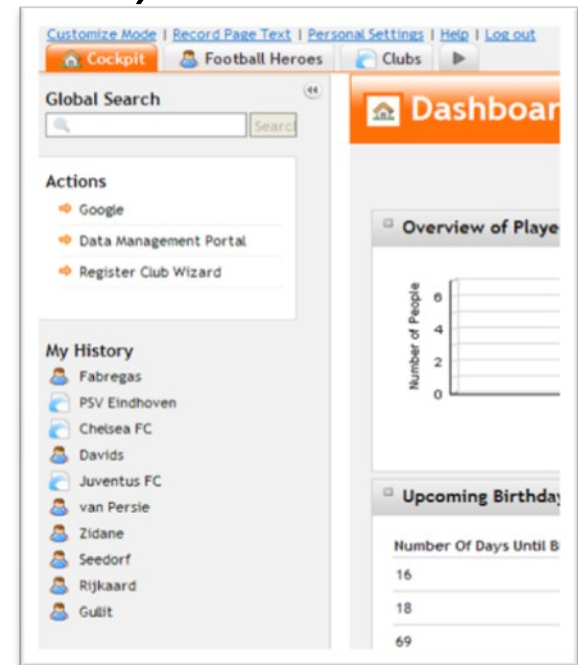


Application Personalization



Personalization Features – My... options

- > My Navigation History
- > My Favorites (quick links, bookmarks)
- > My own custom properties ('flex fields')
- > My Date Format, Weight & Height units, Currency, Default Values
- > My lingo in titles, labels, prompts & messages
- > Personalized Menu-structure
- > ... (my logo!)



Demo SaaS style Customization in JSF

- > Run Time Management of Organization specific Resource Bundles
 - Change the boilerplate text defined in the factory
 - Create organization and user specific resources
- > Influencing application behavior through application settings at organization level
 - Configure date format pattern
 - Specify validation parameters
- > Toggle between user/organisation/factory scope



Resource Bundle Management

Select Language
Select Organization
Select User
Level of Customization

English (en)
Classic Videogames Inc.
Mike Charski
Organization

Choose Bundle and Locale

Bundle

Base
Auxilliary Module

Filter Resource Keys by

Locale

English (en)
Dutch (nl)
Spanish (es)
German (de)
French (fr)
Australian English

Query Resources for Locale in Bundle

Manage Resources

Resource Key	OrgId	UserId	Locale	BundleId	ResourceText
<input type="button" value="v"/> <input type="button" value="v"/> <input type="button" value="v"/> <input type="text"/> <div> <div>emptable.hiredateColumn.header</div> <div>emptable.mgrColumn.header</div> <div>emptable.title</div> <div>emptable.commColumn.header</div> <div>emptable.deptnoColumn.header</div> </div>	Classic Videogames Inc. <input type="button" value="v"/>	Mike Charski (Classic Videogames Inc.) <input type="button" value="v"/>	Dutch (nl) <input type="button" value="v"/>	1	
<input type="button" value="v"/> <input type="button" value="v"/> <input type="button" value="v"/> emptable.hiredateColumn.header	No Organization Selected <input type="button" value="v"/>	No User Selected <input type="button" value="v"/>	English (en) <input type="button" value="v"/>	1	Start Date
<input type="button" value="v"/> <input type="button" value="v"/> <input type="button" value="v"/> emptable.enameColumn.header	Classic Videogames Inc. <input type="button" value="v"/>	No User Selected <input type="button" value="v"/>	English (en) <input type="button" value="v"/>	1	Surname
<input type="button" value="v"/> <input type="button" value="v"/> <input type="button" value="v"/> emptable.empnoColumn.header	No Organization Selected <input type="button" value="v"/>	No User Selected <input type="button" value="v"/>	English (en) <input type="button" value="v"/>	1	Identifier
<input type="button" value="v"/> <input type="button" value="v"/> <input type="button" value="v"/> emptable.salColumn.header	No Organization Selected <input type="button" value="v"/>	No User Selected <input type="button" value="v"/>	English (en) <input type="button" value="v"/>	1	Salary
<input type="button" value="v"/> <input type="button" value="v"/> <input type="button" value="v"/> emptable.enameColumn.header	No Organization Selected <input type="button" value="v"/>	No User Selected <input type="button" value="v"/>	English (en) <input type="button" value="v"/>	1	Last Name
<input type="button" value="v"/> <input type="button" value="v"/> <input type="button" value="v"/> emptable.mgrColumn.header	No Organization Selected <input type="button" value="v"/>	No User Selected <input type="button" value="v"/>	English (en) <input type="button" value="v"/>	1	Manager
<input type="button" value="v"/> <input type="button" value="v"/> <input type="button" value="v"/> emptable.title	No Organization Selected <input type="button" value="v"/>	No User Selected <input type="button" value="v"/>	English (en) <input type="button" value="v"/>	1	Overview of Emplo
<input type="button" value="v"/> <input type="button" value="v"/> <input type="button" value="v"/> emptable.deptnoColumn.header	No Organization Selected <input type="button" value="v"/>	No User Selected <input type="button" value="v"/>	English (en) <input type="button" value="v"/>	1	Department



Application Settings Management

Manage Application Settings

Name	Description	DataType	FactoryValue	OverrideLevel	Value	CustomizationDescription
VARIABLE_INCOME	Support for variable income (financial bonuses).	string	yes	Organization	no	This organization does not believe in absur
MAXIMUM_SALARY	What is the highest amount allowed for salaries?	number	999999	Organization	9000	
HIREDATE_IS_UPDATABLE	The hiredate for the employee can be changed (or not)	string	no	Organization		
DATE_FORMAT_PATTERN	What display format should be used for date values	string	MM/dd/yyyy	Individual User	yes	
HIGH_SALARIES_BACKGROUND_COLOR	Which color should be used to mark high salaries?	string	yellow	Individual User	no	
HIGH_SALARY_LIMIT	What is the threshold value for high salaries?	number	3000	Organization		

[Cancel](#)
[Save](#)

[Employee Form](#)

[Employees Table](#)

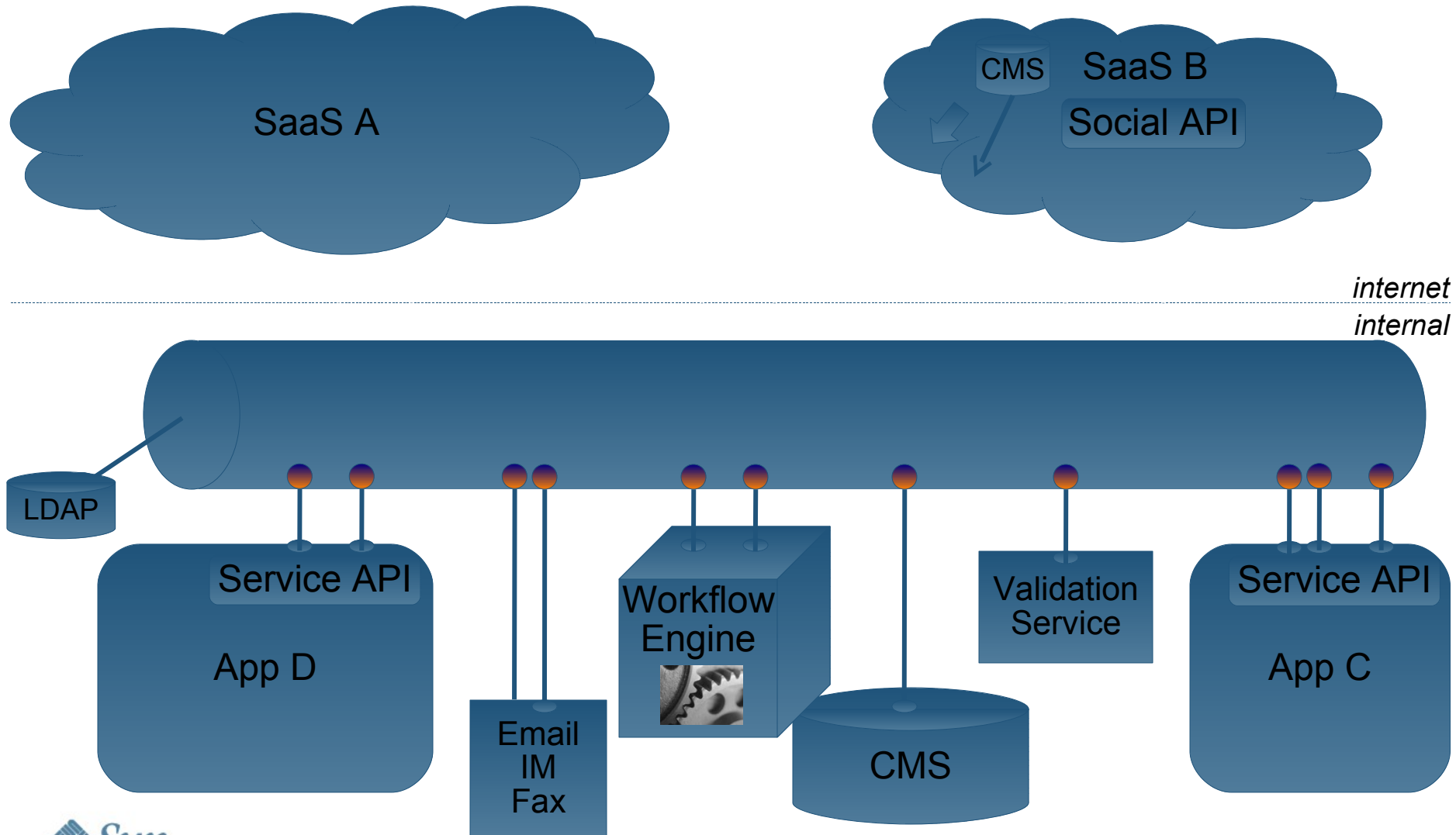
[Department Form](#)

[D](#)


Overview of Employees

Identifier	Surname	Position or Role	Manager	Start Date	Salary	Department
7369	SMITH	BOSS	7902	12/17/1981	1600	20
7499	ALLEN	SALESMAN	7698	02/20/1981	1765	30
7521	WARD	SALESMAN	7698	02/22/1981	1375	30
7566	JONES	MANAGER	7839	04/02/1981	3272.5	20
7654	MARTINA	SALESMAN	7698	09/28/1981	1375	30
7698	BLAKE	MANAGER	7839	05/01/1981	3135	30
7782	CLARK	MANAGER	7839	06/09/1981	2900	20
7788	SCOTT	ANALYST	7566	12/09/1982	3000	30
7839	KING	PRESIDENT		11/17/1981	5500	10
7844	TURNER	SALESMAN	7409	09/09/1981	1455	30

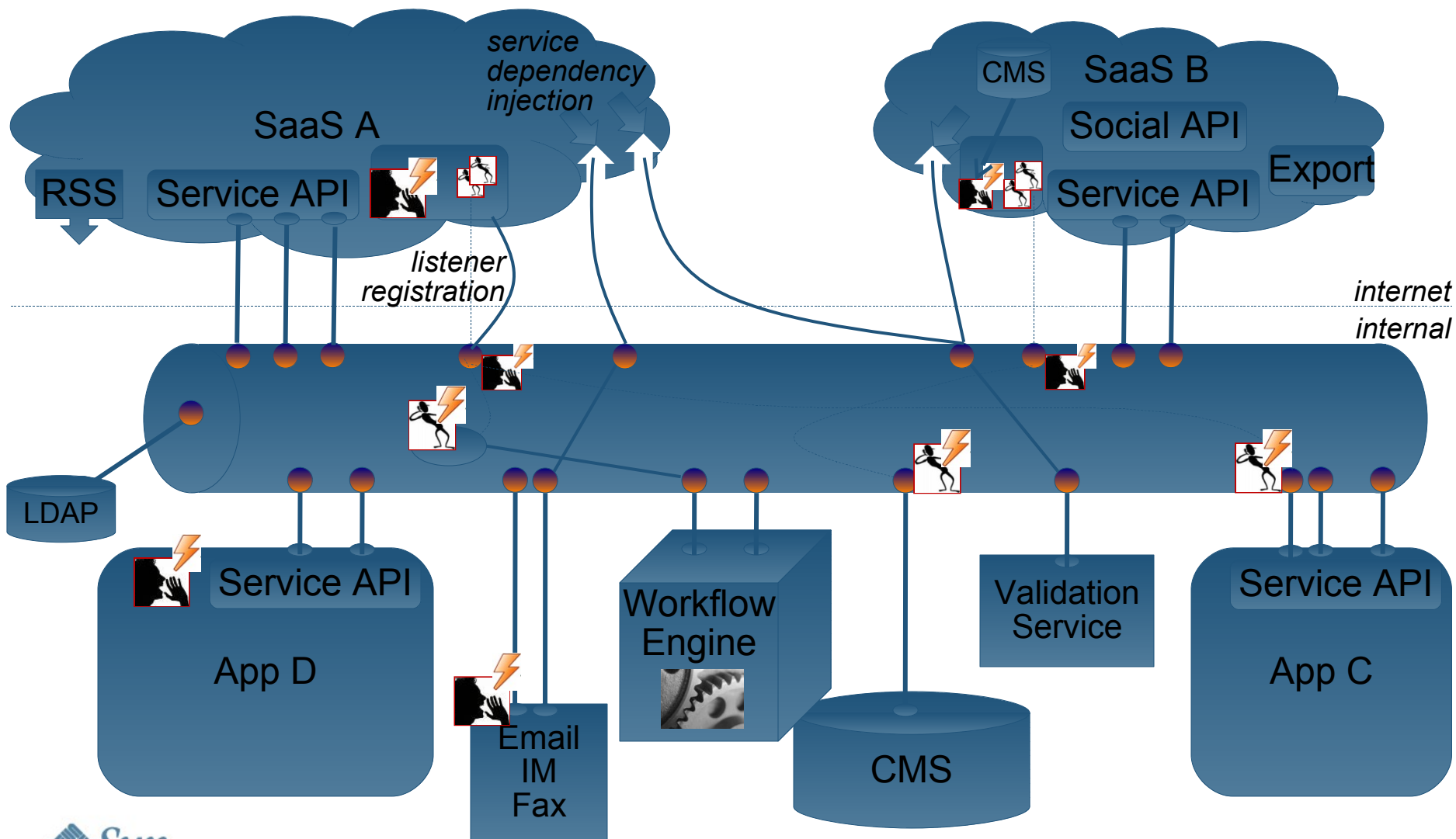
SOA across the cloud



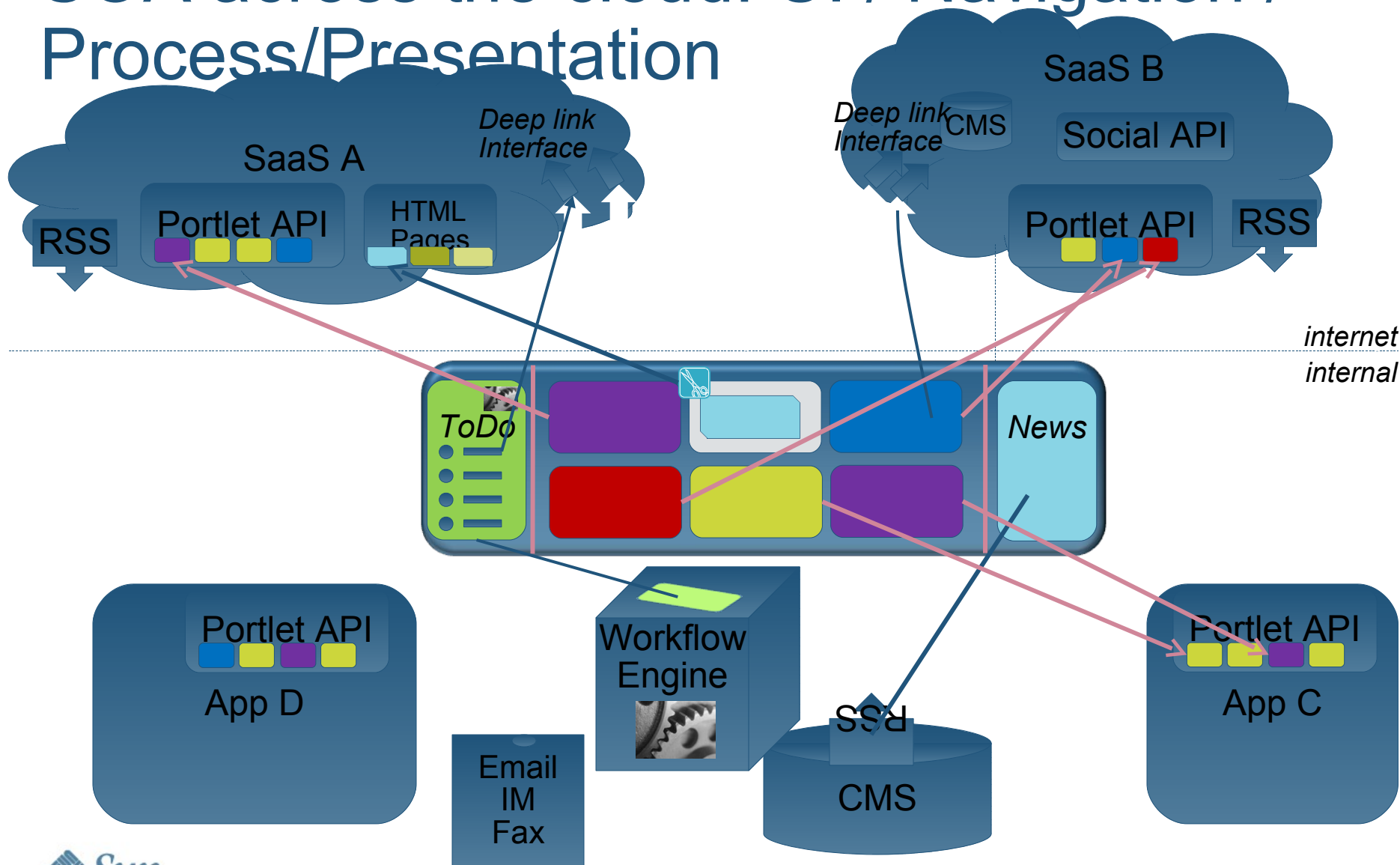
Integration and SOA across the cloud

- > Service API as programmatic alternative to UI
 - RSS, SOAP, REST
- > Event Listener registration to be notified by SaaS
- > Service Registration (Dependency Injection)
 - Authentication/Authorization, Translation, Validation, Workflow, Communication
- > Identity cross referencing (SaaS & local identifier)
- > Deep Link API for context sensitive navigation
- > UI Services (Portlets) to fit into Mashups & Portals
- > Data Export, Synch & Import – bulk facilities
- > “internet service bus” – integrating multiple SaaSs

SOA across the cloud: data & operations

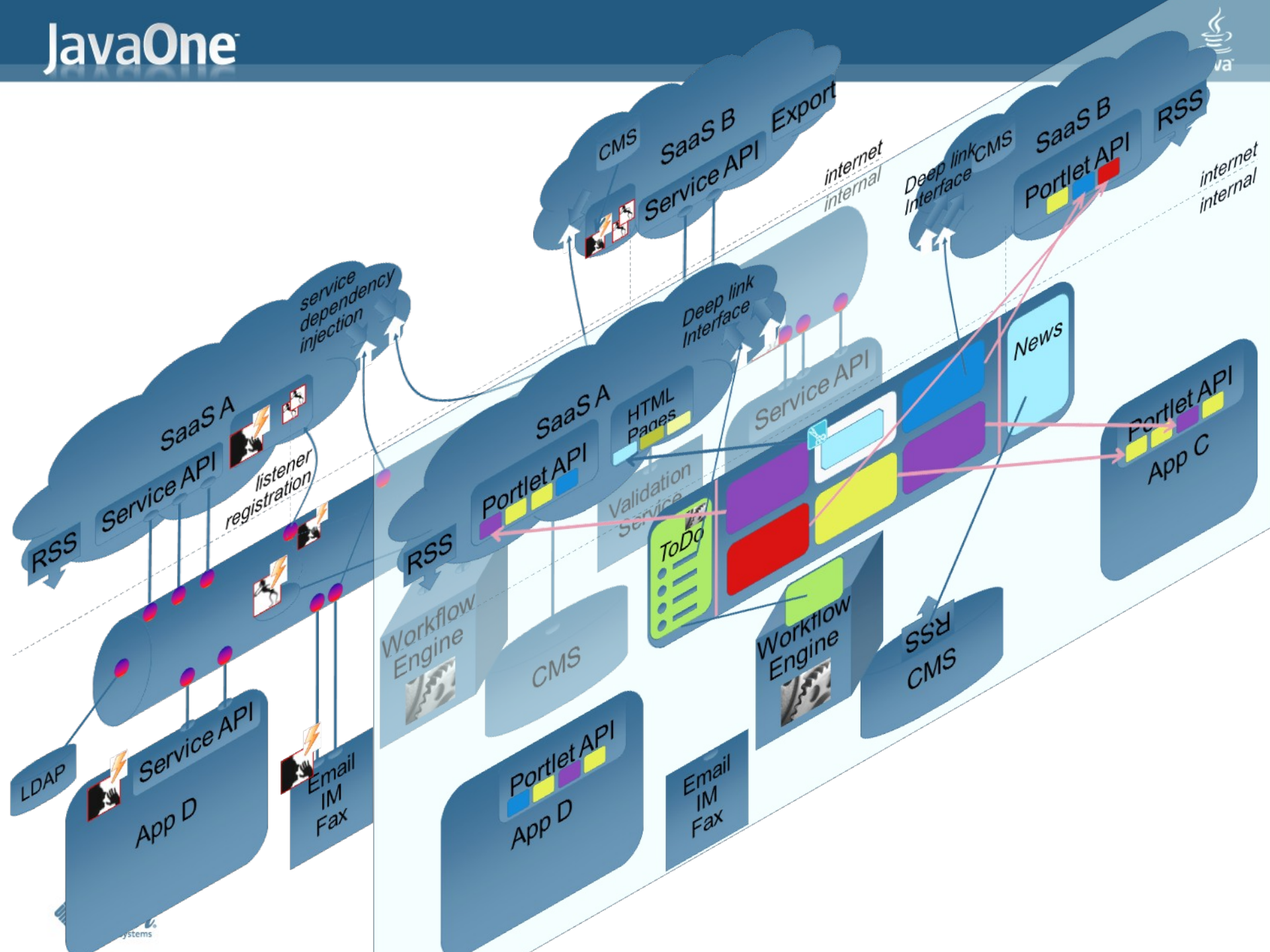


SOA across the cloud: UI / Navigation / Process/Presentation

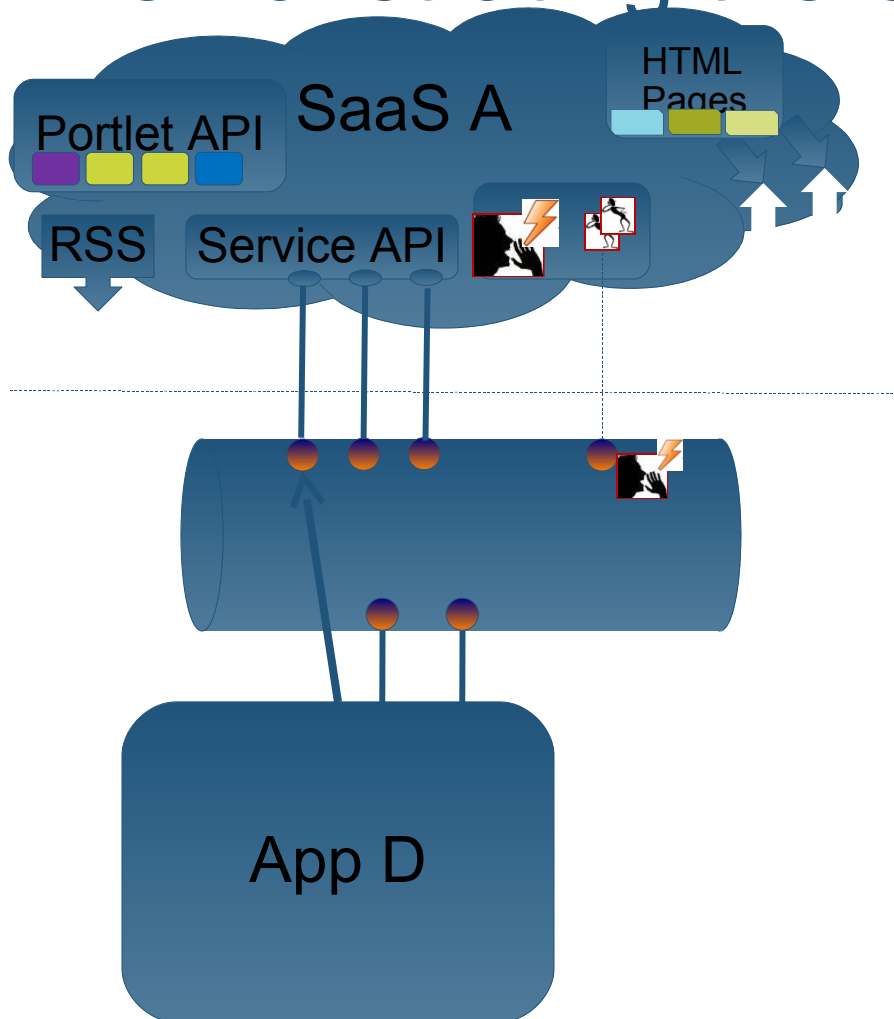


Implementing Deep Linking in JSF

- > URL like <http://host/SaaSApp/UpdateJob/Scott>
 - Cookie provides hook for Single Sign On
- > ServletFilter intercepts and redirects to proper JSF page – after storing parameters in the request
- > JSF PhaseListener intercepts the request to the page and prepares the backing beans
 - Using the request parameters
- > JSF page is rendered with proper data context
- > When the user is done, the JSF application could
 - update the task by calling the workflow engine SAPI
 - publish a SaaS event

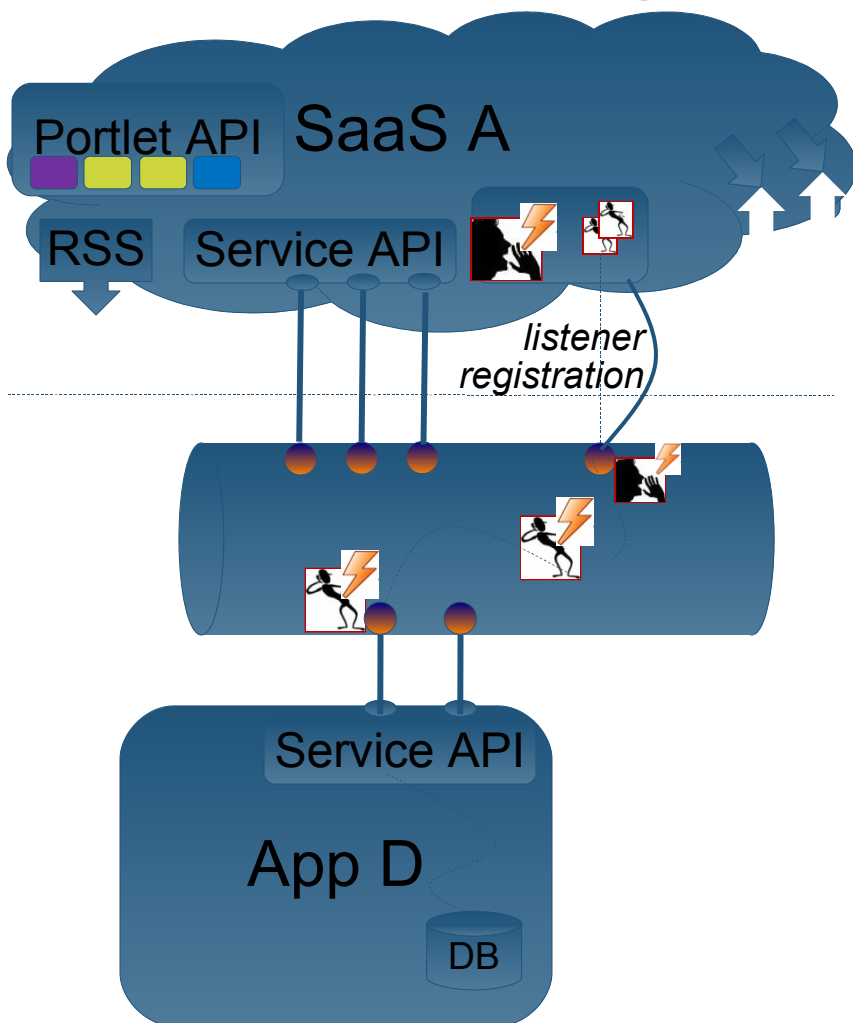


Demonstrating the SaaS Service API



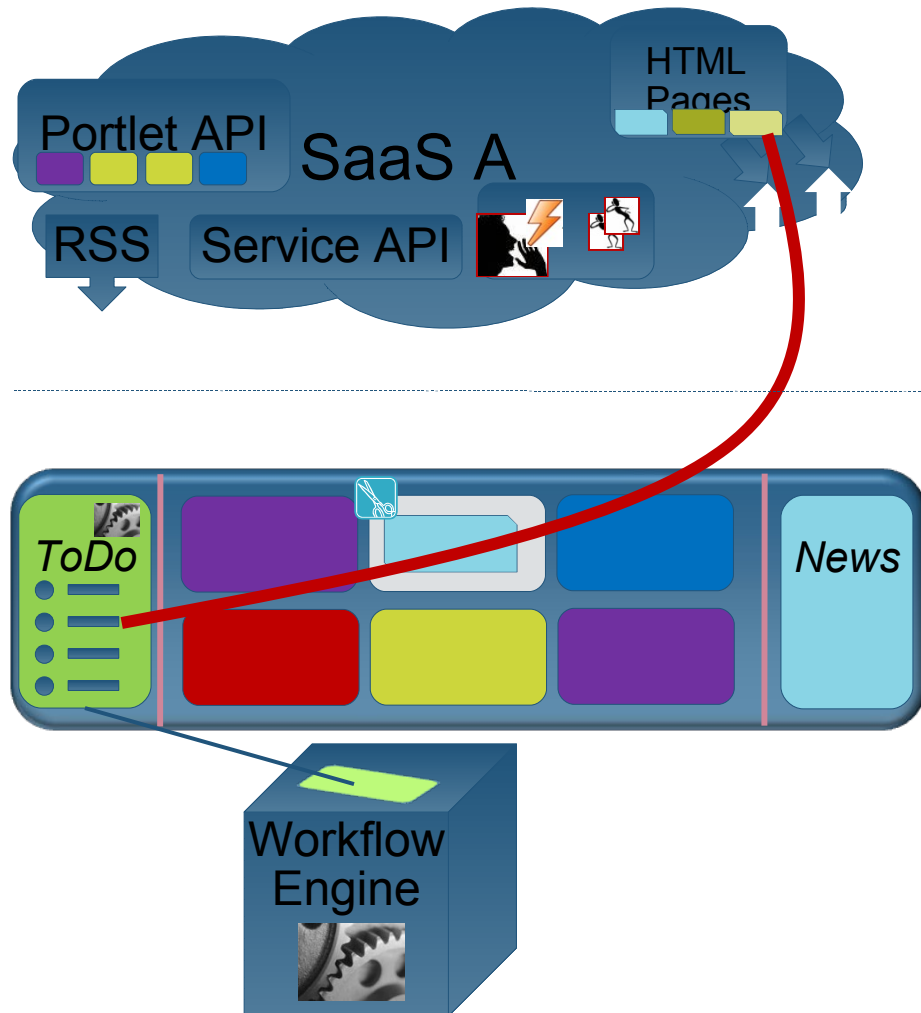
- > Create a new Employee
- > Call the SaaS API to propagate the new employee record
 - Right after successful transaction commit
- > Inspect the new record in the SaaS UI

Demonstrating cross-cloud event listeners

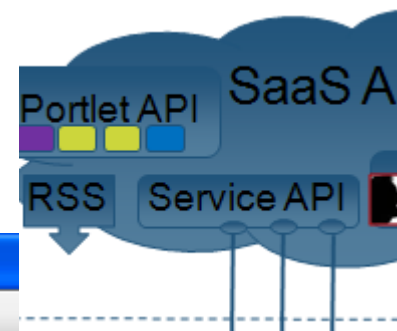


- > Register App D's Update Employee service as Event Listener for SaaS event
- > Update a salary in the SaaS UI
- > Event is routed to the remote listener that calls the App D Service API that "does something with it"

Demonstrating Deep Link Navigation



- > To Do list shows task “Handle promotion for Employee Scott”
- > Selecting this task will bring up a popup loaded through a deep link into SaaS A
 - > Presenting the ‘Update Job’ page with the details of Scott loaded



soapUI 2.0

File Tools Desktop Help



Navigator

Projects

- HrmEmployeeWebService
 - HrmEmployeeWebServiceAPISoapHttp
 - createEmployee
 - getEmployeeDetails
 - Request 1
 - removeEmployee
 - updateEmployee
 - PurchasingPrices
 - RegisterSaaSEventListener
 - SaaSEventListenerRegistrationServiceS
 - deregisterSaaSEventListener
 - registerSaaSEventListener
 - Request 1
 - SaaSEventListenerWebService_testTheList
 - MySaaSEventProcessingServiceSoapHttp
 - handleSaaSEvent
 - Request 1
 - Request 2
 - SimplePatientDataService

Request 1

http://14-317-265:8988/ADFFaces_CustomizableResourceBundles-WebServiceAPI-context-root/HrmEmployeeWeb

Raw XML

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <typ:getEmployeeDetailsElement>
      <typ:empId>7369</typ:empId>
    </typ:getEmployeeDetailsElement>
  </soapenv:Body>
</soapenv:Envelope>
```

Raw XML

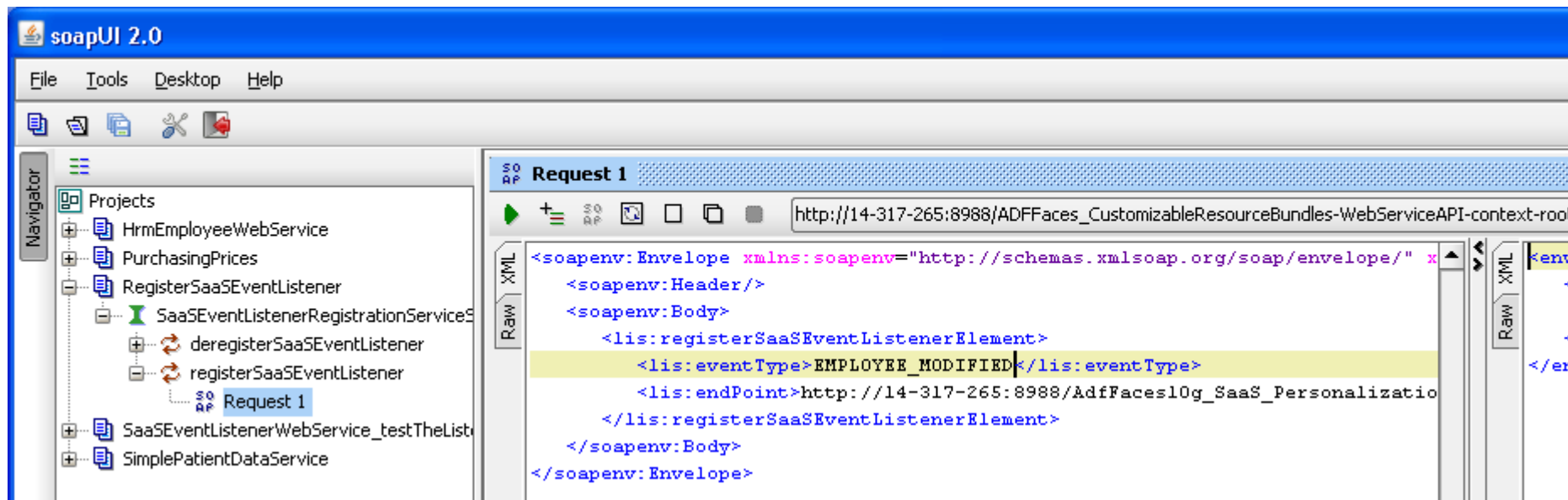
```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Body>
    <ns0:getEmployeeDetailsResponseElement>
      <ns0:result>
        <ns0:managerId>7902</ns0:managerId>
        <ns0:jobTitle>BOSS</ns0:jobTitle>
        <ns0:hiredate>1981-12-17T00:00:00.000+01:00</ns0:hiredate>
        <ns0:bonus>100</ns0:bonus>
        <ns0:name>SMITH</ns0:name>
        <ns0:id>7369</ns0:id>
        <ns0:salary>1600</ns0:salary>
        <ns0:departmentIdentifier>20</ns0:departmentIdentifier>
      </ns0:result>
    </ns0:getEmployeeDetailsResponseElement>
  </env:Body>
</env:Envelope>
```

Aut Headers (0) Attachments (0)

response time: 9942ms (737 bytes)

Headers (8) Attachments (0) SSL Info WSS (0)

Register Event Listener by calling the WebService



The screenshot displays the soapUI 2.0 application window. The title bar reads "soapUI 2.0". The menu bar includes "File", "Tools", "Desktop", and "Help". Below the menu bar is a toolbar with icons for file operations and testing. The left sidebar, labeled "Navigator", shows a tree of projects. The "RegisterSaaSEventListener" project is expanded, revealing a sub-project "SaaSEventListenerRegistrationServiceS" which contains two test cases: "deregisterSaaSEventListener" and "registerSaaSEventListener". The "registerSaaSEventListener" test case is selected, and its "Request 1" is highlighted. The main pane shows the XML content of "Request 1" in a raw view. The URL bar at the top of the main pane shows the endpoint: "http://14-317-265:8988/ADFFaces_CustomizableResourceBundles-WebServiceAPI-context-root". The XML content is as follows:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" x
  <soapenv:Header/>
  <soapenv:Body>
    <lis:registerSaaSEventListenerElement>
      <lis:eventType>EMPLOYEE_MODIFIED</lis:eventType>
      <lis:endPoint>http://14-317-265:8988/AdfFaces10g_SaaS_Personalizatio
    </lis:registerSaaSEventListenerElement>
  </soapenv:Body>
</soapenv:Envelope>
```

Employee Form
Employees Table
Department Form
Departments Table

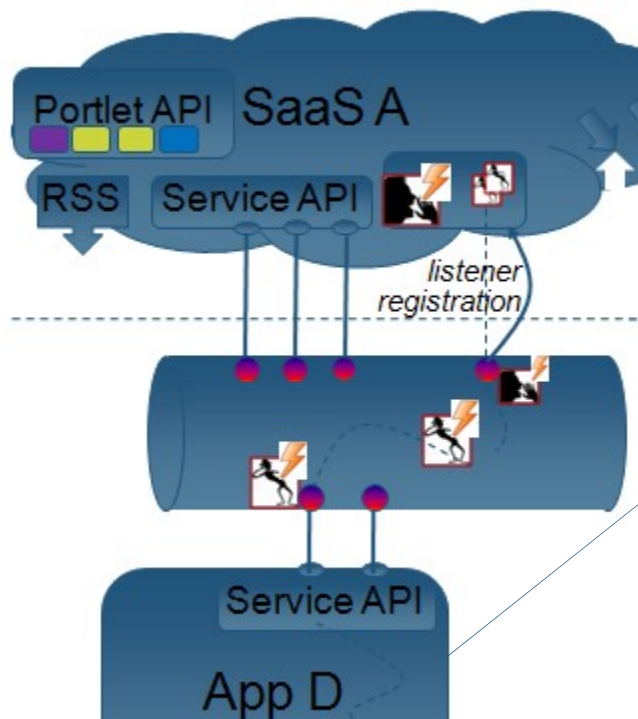
Employee

Edit Employee
[New](#)

Empno	7369	Sal	1800
Ename	SMITH	Deptno	30
Job	MANAGER		
Mgr	7902		
Hiredate	12/27/1983		

[Previous](#)
[Next](#)
[Last](#)
[Cancel](#)
[Save](#)

Show All



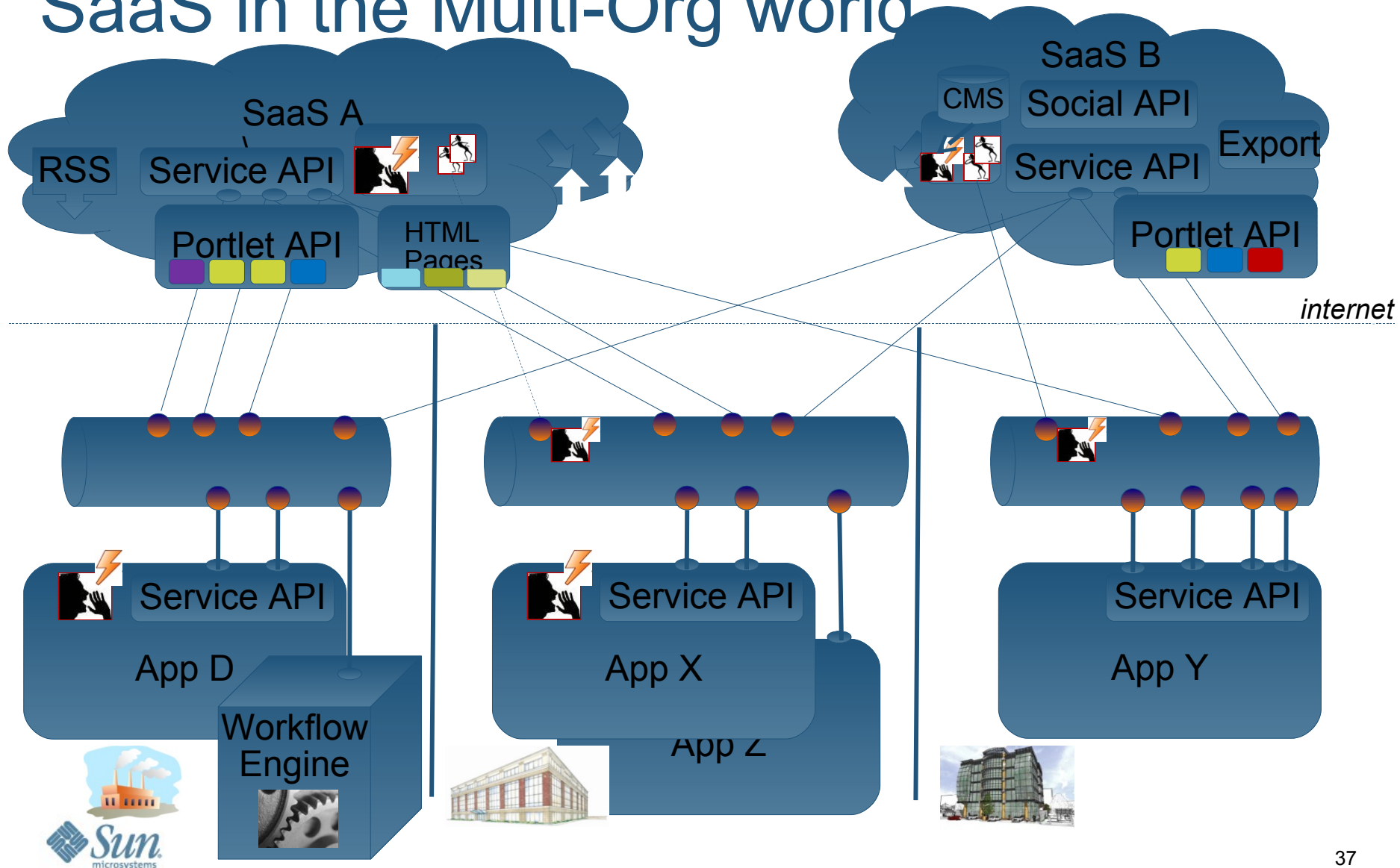
Lucas Jellema

[Call](#)
[Send voicemail](#)
[Send Files](#)
[Email](#)

Lucas Jellema: SaaS Event EMPLOYEE_MODIFIED (id=AMIS_HRM_EMP_Tue Jun 02 01:29:22 CEST 2009) was received was from: SAAS_AMIS_HRM. The event originally took place at 01:29:22 (02-Jun-2009). The payload of the event: An update took place on an employee called SMITH. The changes were: - Job is : MANAGER(was: BOSS); - Sal is : 1800(was: 1600); - Hiredate is : 1983-12-27(was: 1981-12-17); - Deptno is : 30(was: 20);

Sent at 1:29 AM on Tuesday

SaaS in the Multi-Org world



Summary and Conclusion

SaaS and the road ahead

- > SaaS applications need special facilities
 - Customization & personalization
 - Service API (UI and programmatic), Event support
 - SLA management, metrics monitoring & self service
- > Enterprise Service Bus and Portal/Mashup connect the SaaS applications
- > JavaServer Faces can be used for implementing the customizable and integratable front end
- > Most applications could do with the same facilities:
let's regard all applications as SaaS!



JavaOneSM

Thank You

Lucas Jellema

lucasjellema@gmail.com

<http://technology.amis.nl/blog>
(with all source code of the demos)