

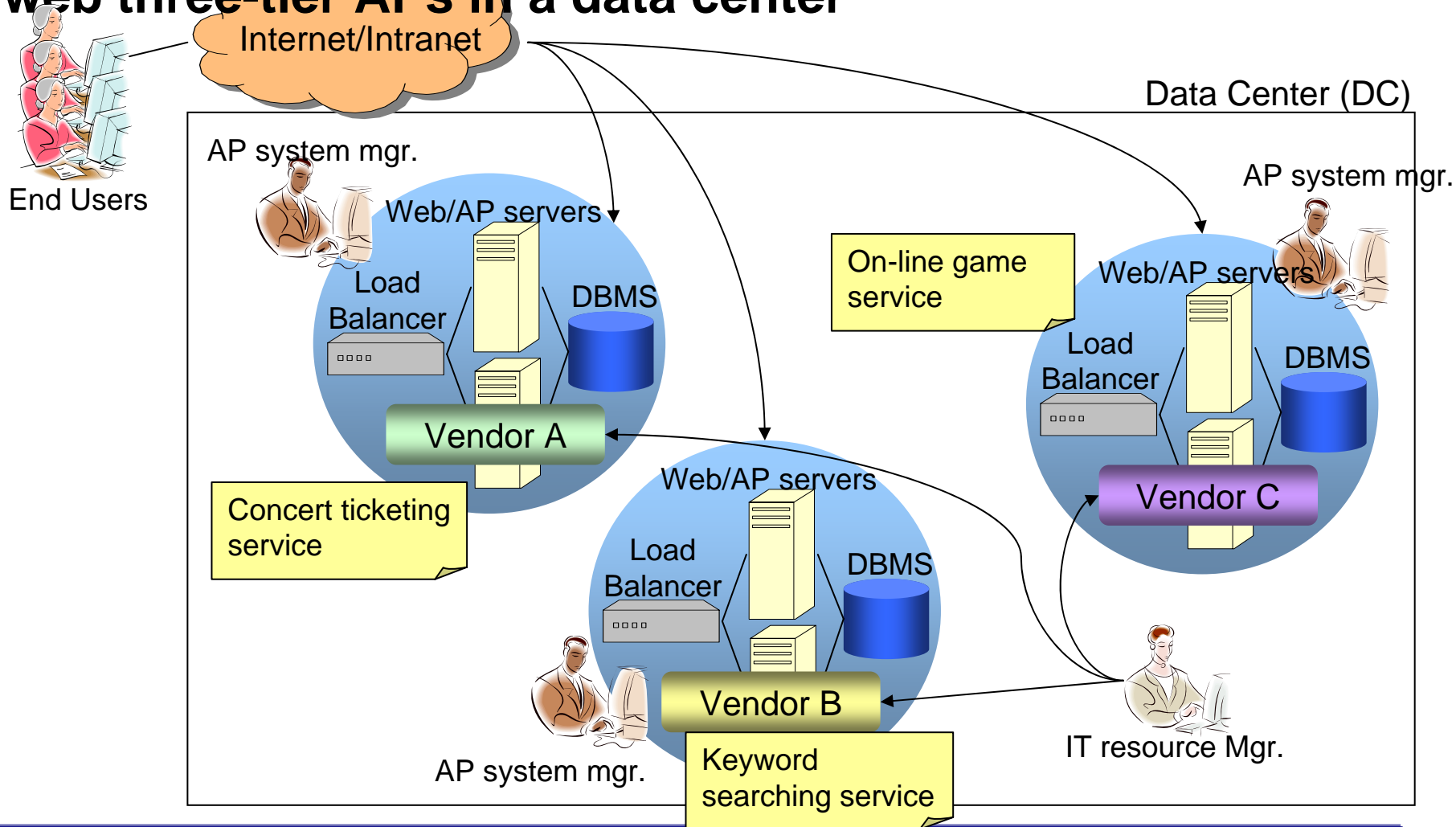
# Use of WS-Agreement in the Business Grid Project

Toshiyuki Nakata (NEC Corp.)  
Shinya Miyakawa (NEC Corp.)  
Andreas Savva (Fujitsu Ltd)  
Hiro Kishimoto (Fujitsu Ltd)  
Nobutoshi Sagawa (Hitachi Ltd)

- Mission: Develop Business Grid middleware
  - Next generation business application infrastructure
  - Contribute to international standardization
- Three year project: 2003 - 2005
- Industry Members: Fujitsu, Hitachi, and NEC
- Collaborate with Grid Technology Research Center of AIST
- Jointly funded by the Ministry of Economy, Trade, and Industry (METI)
- Resultant components are to be available as “open-source”
- For more details please visit EGR-RG 30<sup>th</sup> Jun. 9:00-10:30!!

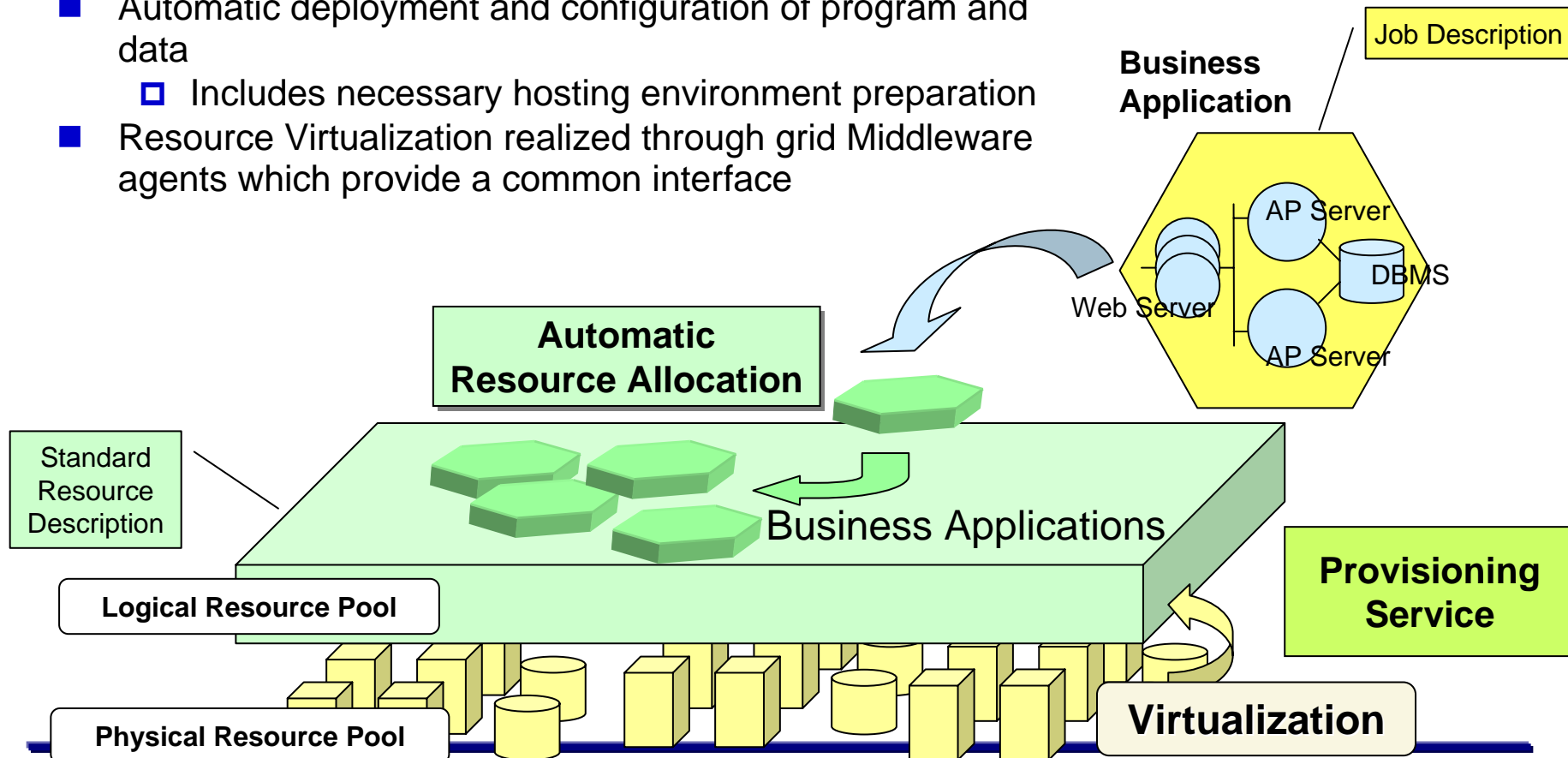
# Application Target

Make it easy to deploy/run business applications such as **web three-tier APs** in a data center

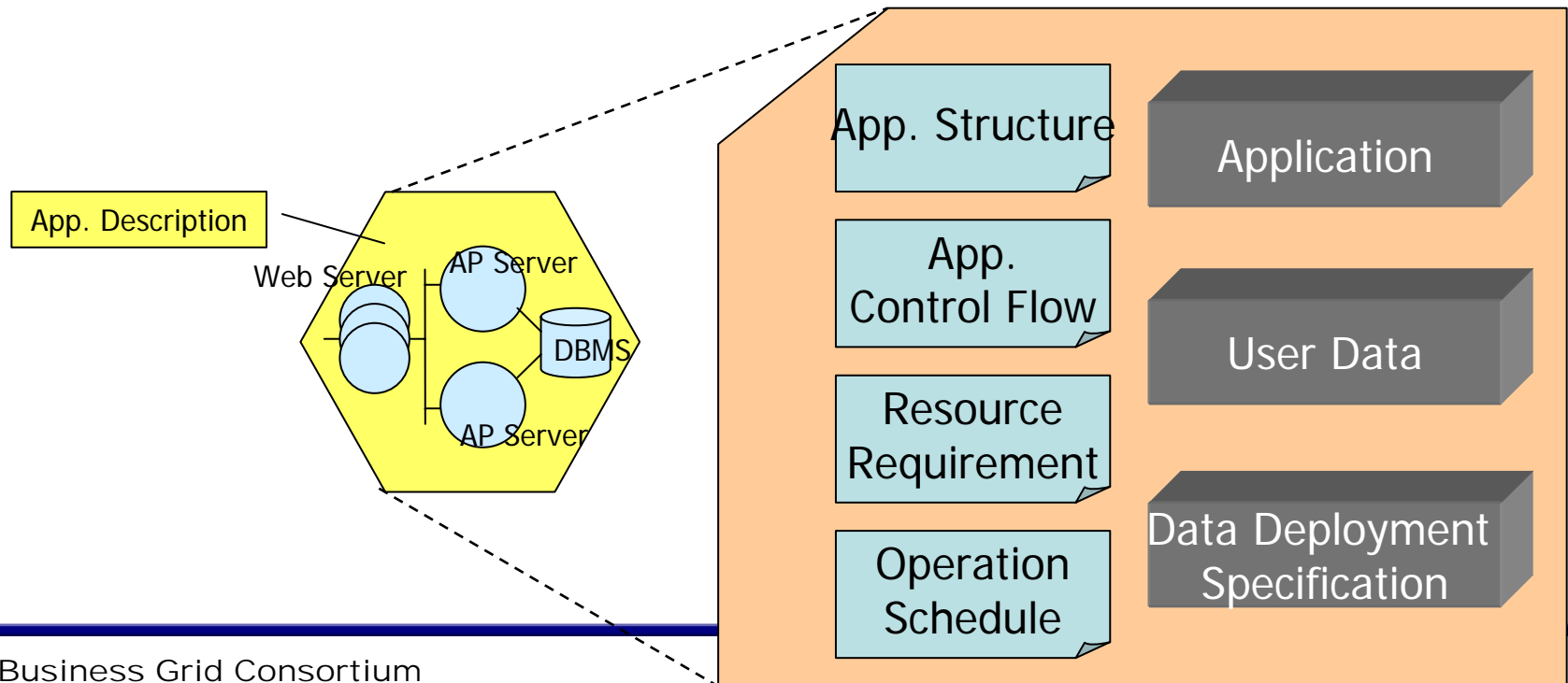


# Big Picture - how it works -

- Job Submission
  - ▣ Standard job description (using WS-Agreement+JSDL with extensions protocol) and application contents service
- Brokering allocates necessary IT resources
- Automatic deployment and configuration of program and data
  - ▣ Includes necessary hosting environment preparation
- Resource Virtualization realized through grid Middleware agents which provide a common interface

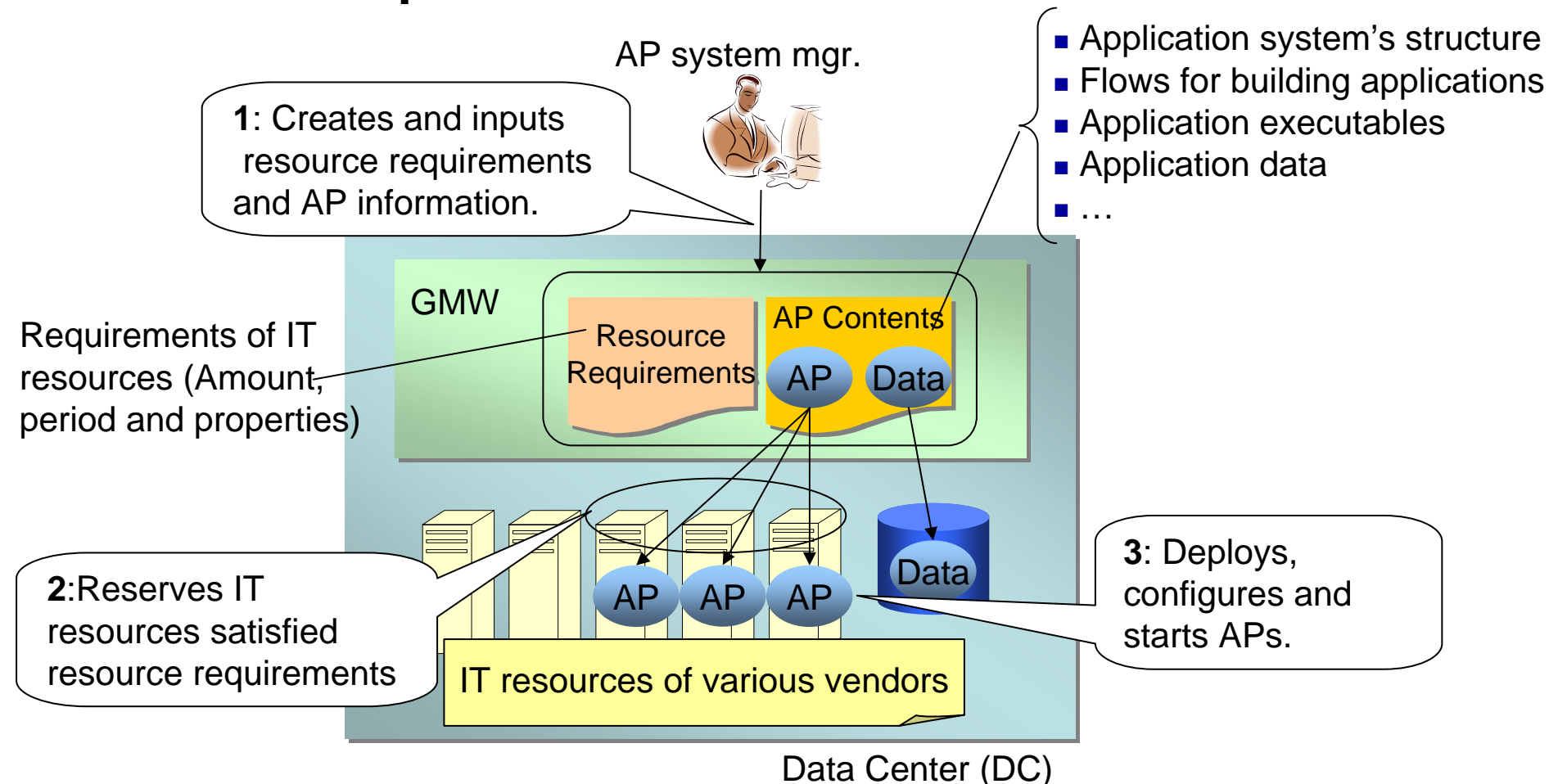


- The Application description in BizGrid not only archives the relevant execution modules, but also maintains all necessary information in one package, in order to manage the entire lifecycle of the operation.
- The description contains the specification of app. structure (e.g. 3-tier Web App). It enables mapping between the job and virtualized resources, automatic deployment of execution modules and autonomic control of the resource allocation.



# Scenario 1 Building an application system

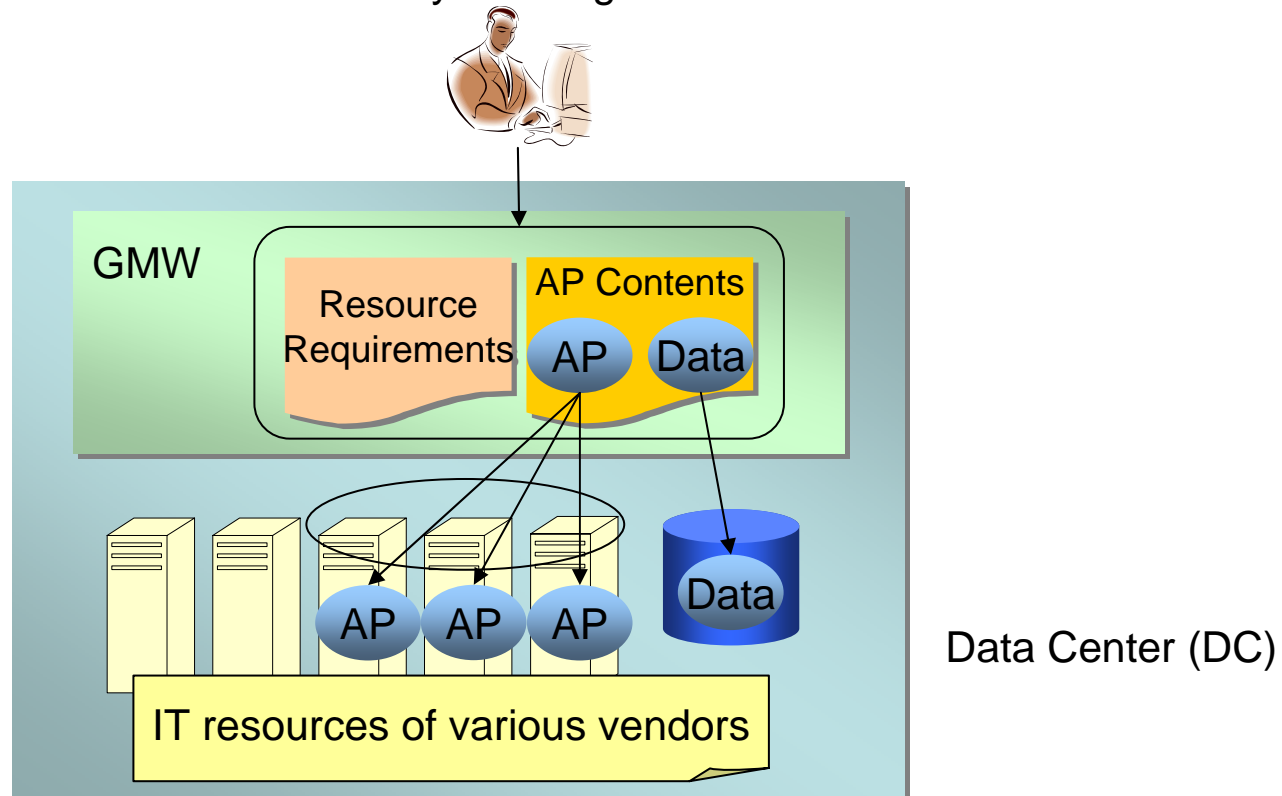
## Building an application system in a data center with resource requirements and AP contents



# Role Players

- Service Provider= Agreement Provider= Data Centers
- Service Client= Agreement Initiator= AP system mgr
  - (End-Users are not player in the scenario)

AP system mgr.



- Domain Specific Language: JSDL (with extensions)
  - JSDL is extended on some areas to support needed functionality (see next slide)
  - Link between JSDL constructs and WS-Agreement:
    - As a convention, contents of JSDL:jobname corresponds to ServiceName

```
<wsag:ServiceDescriptionTerm Name="JOB11Id" ServiceName="JOB11">
  <jSDL:JobIdentification>
    <jSDL:JobName>JOB11</jSDL:JobName>
    <jSDL:JobDescription> This is a BG job that ...
  </jSDL:JobDescription>
  </jSDL:JobIdentification>
</wsag:ServiceDescriptionTerm>
```



## ■ Domain Specific Language: Extension of JSDL

- Why Extension?: To make the description of more complicated resource requirements easier: for example, a Web 3-tier Application which requires various resources such as Load balancers, Web servers, Application Servers and DB servers.

```
<wsag:ServiceDescriptionTerm Name="GPTforJOB11" ServiceName="JOB11">
  <jSDL:Resources>
    <bg:poolTypes>
      <bg:poolType kind="Web3Tier">
        <bg:resource type="Web"/>
        <bg:resource type="LB"/>
        <bg:resource type="AS"/>
        <bg:resource type="DB"/>
      </bg:poolType>
    </bg:poolTypes>
  </jSDL:Resources>
</wsag:ServiceDescriptionTerm>
```

Tentative!! & some  
Of the tns, contents  
Modified for ease of reading

- Domain Specific Language: Extension of JSDL (cont'd)
  - For example to describe reservations

```
<wsag:ServiceDescriptionTerm Name="JOB10_Reserv." ServiceName="JOB10">
  <jSDL:Resources>
    ...
    <bg:Reservation type="fixed"/>
      <bg:ReservationBeginTime dateTime=
        "2004-11-01T09:00:00.000+09:00"/>
      <bg:ReservationEndTime dateTime=
        "2004-12-01T09:00:00.000+09:00"/>
    </bg:Reservation>
  </jSDL:Resources>
</wsag:ServiceDescriptionTerm>
```

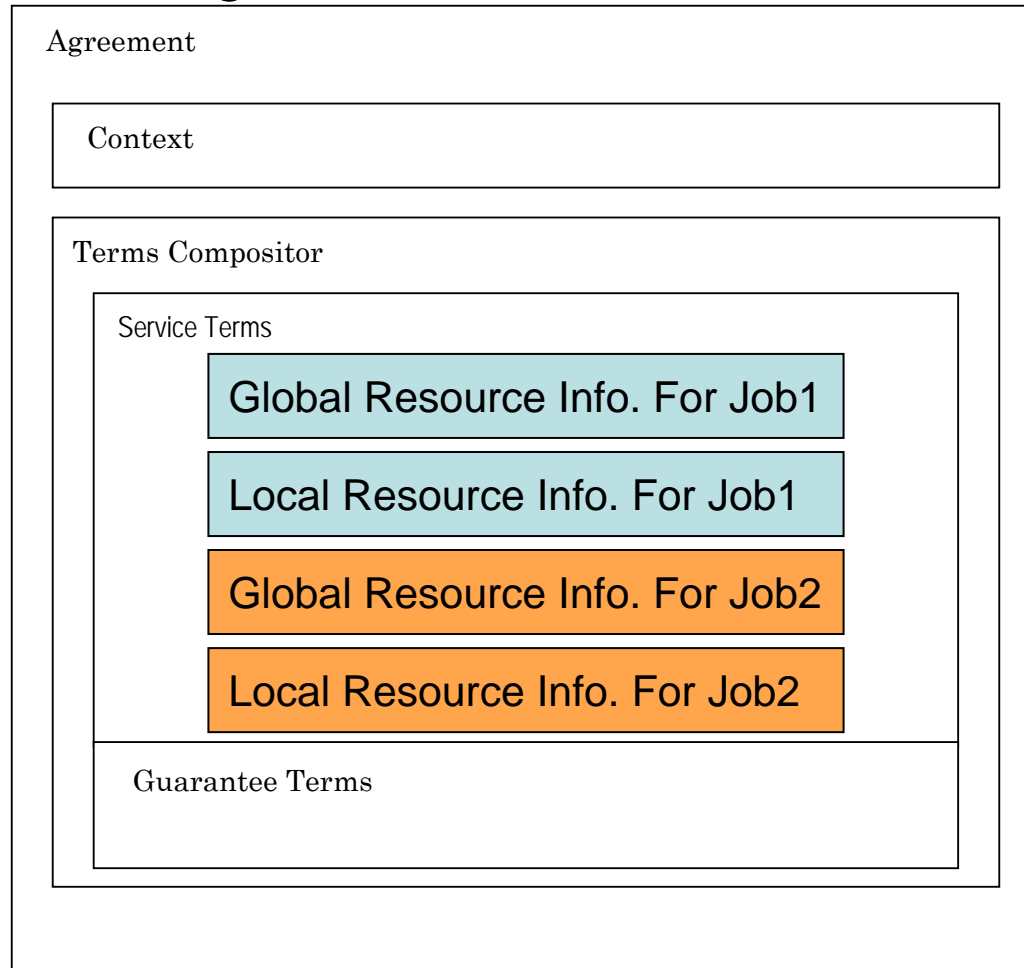
Tentative!! & some of the tns, contents  
Modified for ease of reading

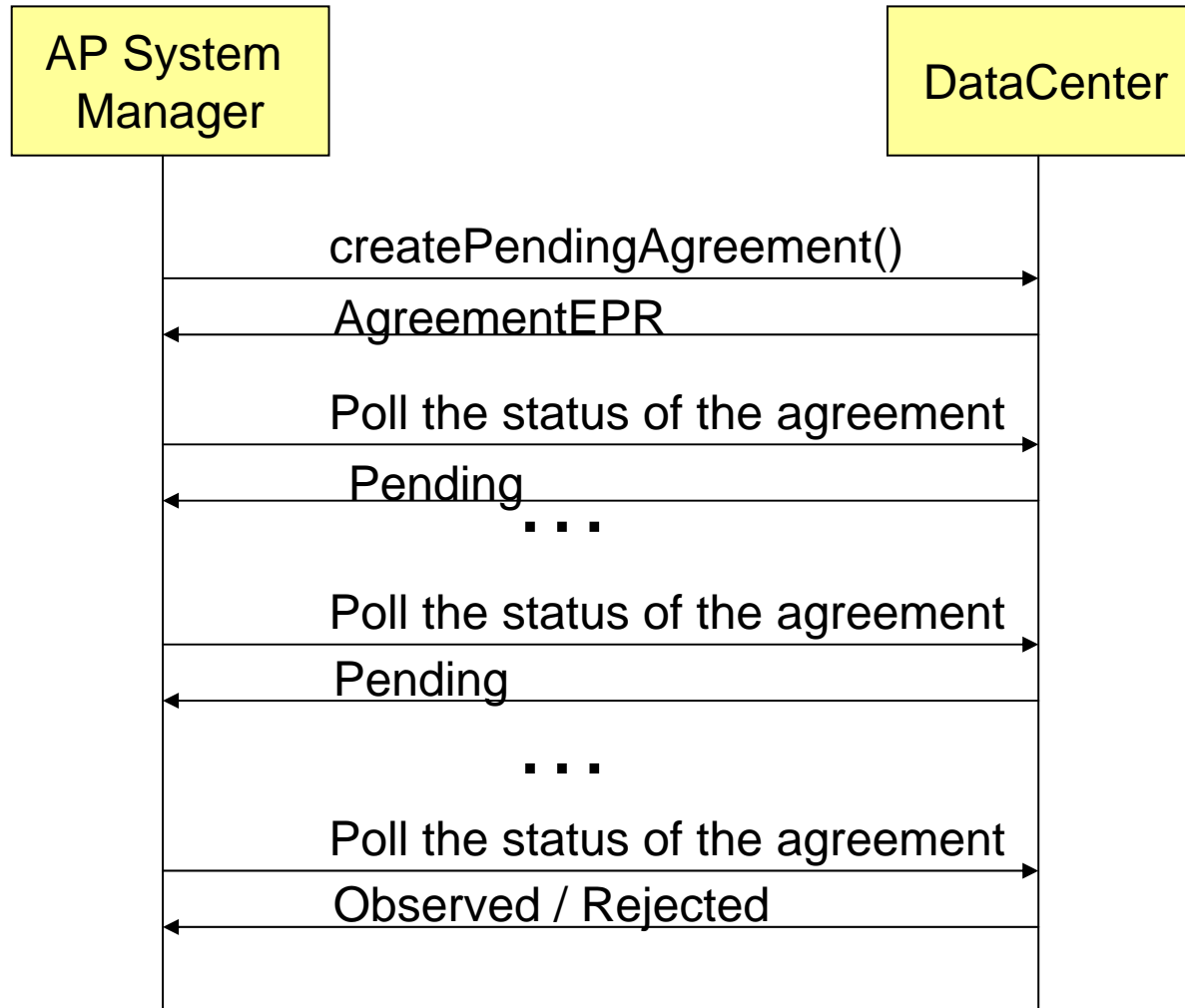
- Two types of resource description information for each Job:
  - 'Global' resource info. for describing the type of Job:
    - Eg. A Web3tier AP which consists of a Load balancer, Web server or a batch job..
    - General characteristics of the job (To allow for automatic load control or not.)
  - 'Local' Resource info. which describes the resource needed for each of the components that make up a pool to carry out the job.

```
<wsag:ServiceDescriptionTerm Name="GLBforJOB10" ServiceName="JOB10">
  <jSDL:Resources>
    <jSDL:TotalResourceCount>
      <exact>1.0</exact>
    </jSDL:TotalResourceCount>
    <!-- BG specific things deleted ...-->
  </jSDL:Resources>
</wsag:ServiceDescriptionTerm>
```

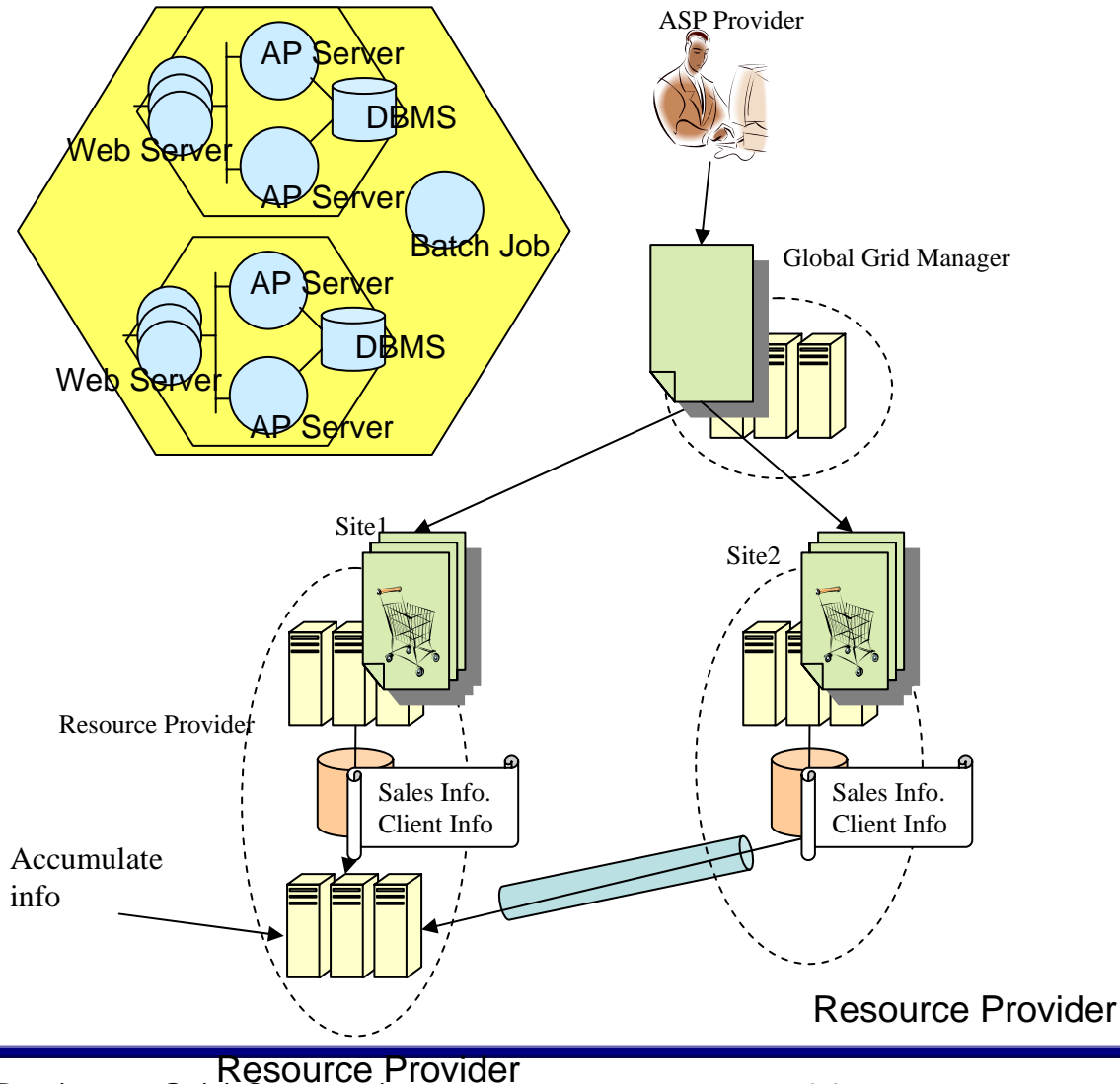
- Several jobs which are related such as a Web3Tier AP for a web-shop and a batch job which calculates the sales info every weekend can be included in a single agreement. (In fact at this time without the use of related agreements they would have to be..)

- Uses mostly context, service terms and might use a little bit of guarantee terms.



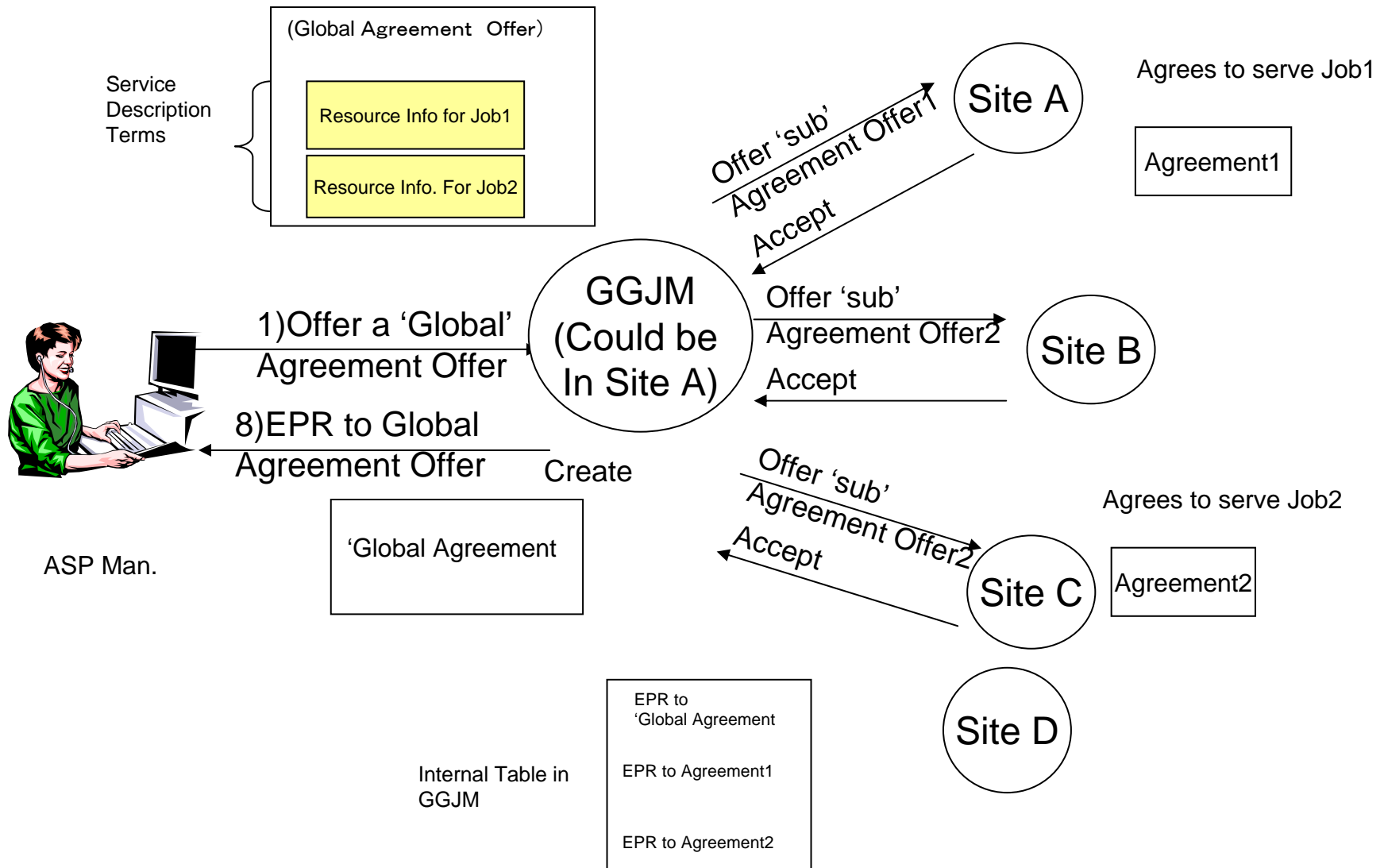


## JobDescription

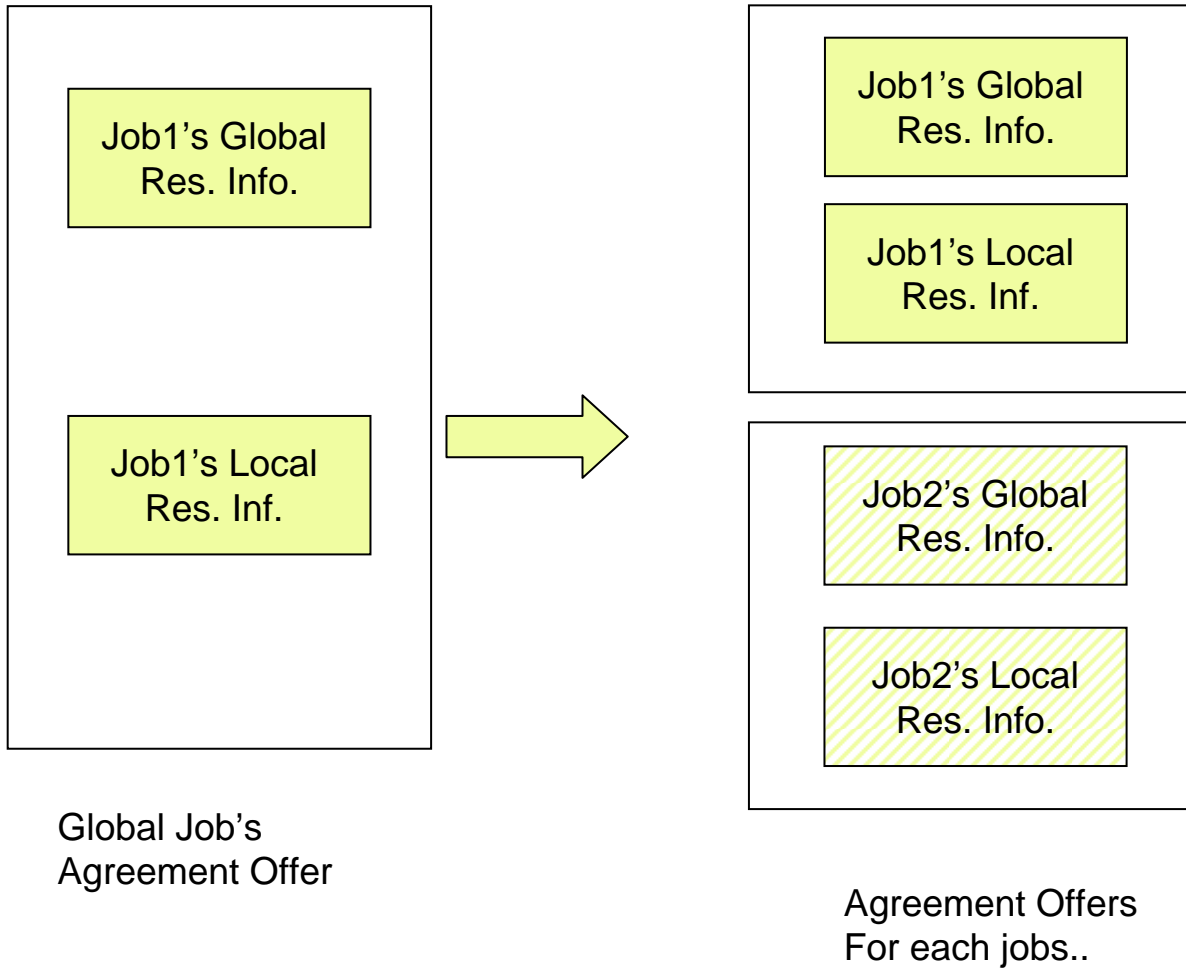


Share IT resources based on the contract/agreement among 1) Distributed Centers in an Enterprise, 2) Among Trusted partner Data Centers => Make it possible for an ASP Provider (client) to dispatch a Complex Job from an entry point

# Image of Global Agreement Offers and the sequence

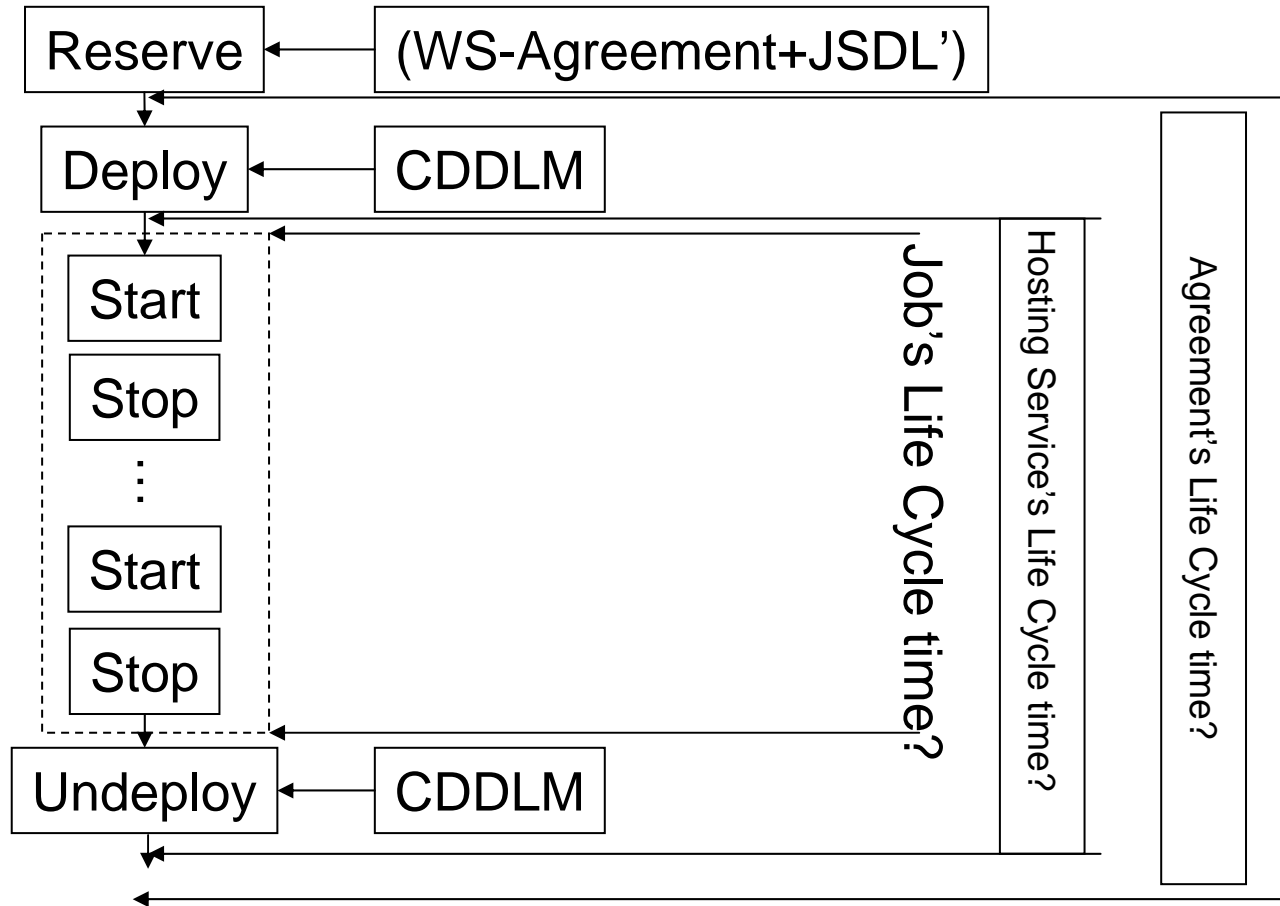


# Splitting the offer by GGJM





# Lifetime of Agreement/Services etc



# Wishlists (As input for 30th's discussion)

---

- To make it possible for the Agreement Provider to modify the Agreement in order to give more info to the Agreement Initiator (This if agreed would be in the AgreementNegotiation part)
  - Eg. Prune the  
    <wsag:ExactlyOne>  
        wsag:TermCompositorType  
    </wsag:ExactlyOne>  
    To the one actually chosen.
  - Modify the AgreementExpirationTime based on its ability to observe the Agreement
  - Return an EPR of RelatedAgreements which were created in ordered to meet the original RelatedAgreement
- Resurrection of RelatedAgreements
  - Make it possible to create a new batch job which was found necessary related to previously agreed and currently running jobs.
  - As a place holder for 'future' jobs (sub jobs in the previous example)
- Linkage with CDDLML
  - For more complex jobs, info. for how systems are deployed might be needed (A bit domain specific?)