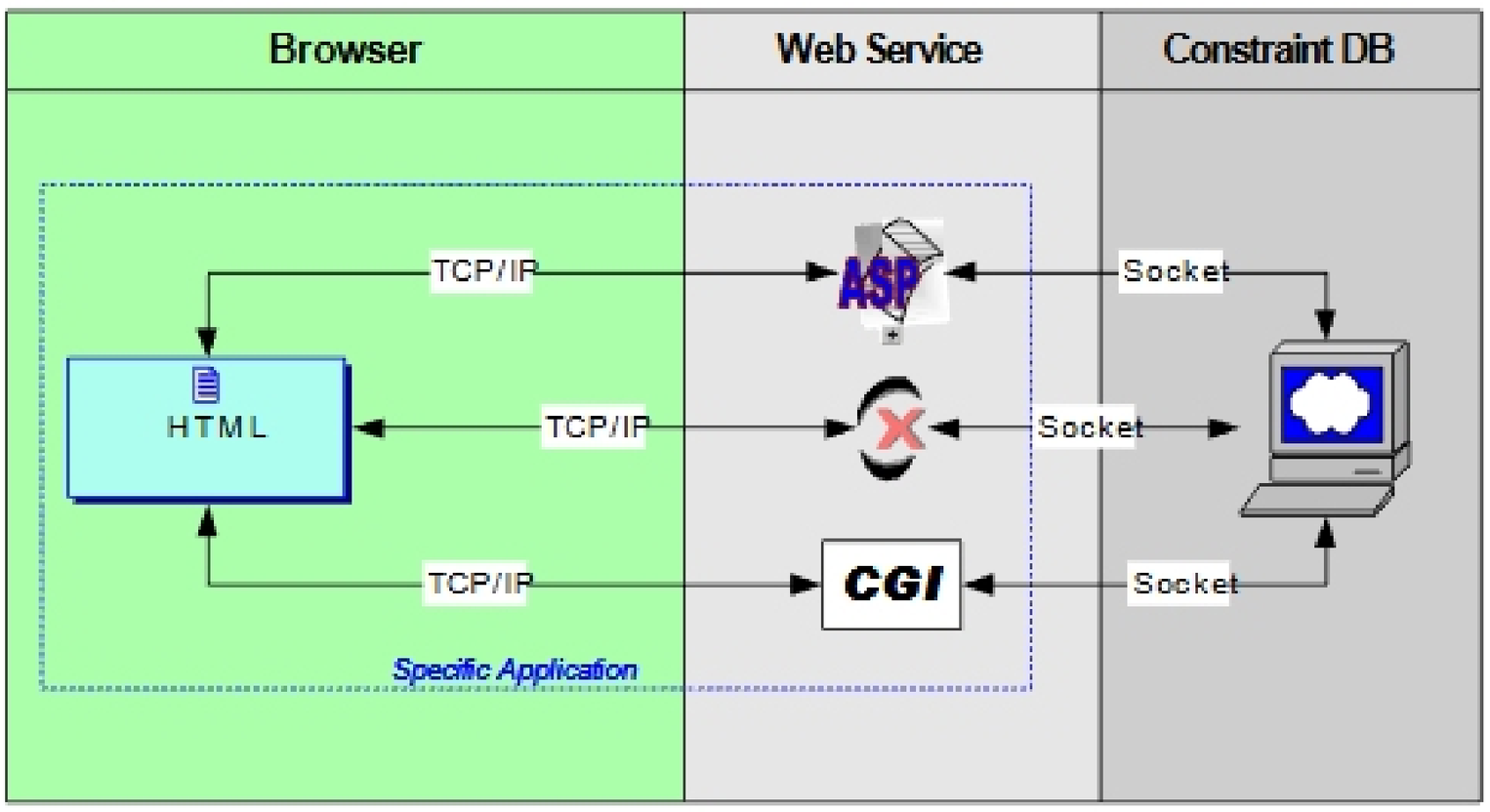
Implementing Web Applications in MLPQ System

Shasha Wu (shwu@cse.unl.edu)

I. Designing web applications in MLPQ System

1. Designing Infrastructure.



2. Responsibilities for three tiers:

Browser	Web Service	CDB
• User interfaces. • Send user's requests to Web Server. • Accept results from Web Service. • Display the results.	• Accept requests from the browser. • Parse the request to CDB acceptable command and arguments. • Send commands to CDB. • Accept result from CDB. • Explain the result (Optional) • Return result to Browser.	• Accept commands and arguments from Web Server. • Activate related operations according to the command and arguments. • Return the results to Web Service.

3. Commands and arguments for CDB:

Command	Arguments	Actions on CDB	Result

Open	<user> <filename>$	Create a view for the user and open a CDB data file.	0 (success) 1 (fail)
Close	<user> <filename>$	Close the view of the user.	0 (success) 1 (fail)
SQLBasic	<user> <filename> <relation name> #<select>#<from>#<where>$	Execute a SQL query on the user's view.	File.txt# 1 (fail)
SQLAggregate	<user> <filename> <relation name> #<select>#<from>#<where> #<group>#<having>$	Execute a SQL query on the user's view.	File.txt# 1 (fail)
SQLSet	<user> <filename> < relation name> #<select1>#<from1>#<where1> #<set_op> #<select2>#<from2>#<where2>$	Execute a SQL query on the user's view.	File.txt# 1 (fail)
SQLNested	<user> <filename> <relation name> #<select1>#<from1>#<where1> #<nest_op> #<select2>#<from2>#<where2>$	Execute a SQL query on the user's view.	File.txt# 1 (fail)
Datalog	<user> <filename> <datalog string>$	Execute a Datalog query on the user's view.	file.txt# 1 (fail)
Include	<user> <filename> <relation name> <color>$	Highlight the relation by its name and assign a color for it in the view.	0 (success) 1 (fail)
Clear	<user> <filename>$	Deselect all relations	0 (success) 1 (fail)
GetImage	<user> <filename>$	Copy the image of the view in screen and save to disk.	File.bmp# 1: fail

ColorRelation	<user> <filename> <relation_name>$	Do color relation on given relation.	
Zoom	<user> <filename> <x> <y> <w> <h>$	Zoom the image.	
GetAnimation	<user> <filename> <Start> <End> <Step>$	Generate serious animation images	

*color: 0=black; 1=red; 2=green; 3=blue.

The default environment settings are saved in mlpq_nt.cfg file and the values of the settings are displayed in following table. The system administrator can modify these settings in the file to configure the server.

Name and Value	Description
WorkDir = "c:\inetpub\wwwroot\dbs\",	Define the directory to load constraint database files by the server.
OutPut = "c:\inetpub\wwwroot\dbs\",	Define the directory for the server to output its result file.
Port = "2222".	Define the value of socket port

4. Sample codes in middle tier:

```
Open  Sam  Police.txt$
SQLBasic  Sam  Police.txt  view1#Contains.Street, Contains.x, Contains.y#
      Contains# Contains.Street= "Vine"$
Datalog  Sam  Police.txt  view2(id, x, y):- Contains(id, x, y), id="Oak"$
Include  Sam  Police.txt  Contains 0$
Include  Sam  Police.txt  view1 2$
Include  Sam  Police.txt  view2 1$
GetImage  Sam  Police.txt$
ColorRelation  Sam  test6.txt  test6$
Zoom  Sam  test6.txt  300  20  100  60$
GetAnimation  Sam  torpedo.txt  0  25  1$
```