Dynamic Workload Management using CICSPlex System Manager

Dan Donnelly

Donnelly@uk.ibm.com

Session 4139A

impact·venture

CICSPlex SM Review

- Enterprise systems management for CICS Transaction Server via:
  - Web User Interface
  - Batch repository update
  - Application Programming Interface

- Single System Image & Single Point of Control for:
  - Resource definition
  - Operations
  - Monitoring
  - Threshold Analysis
  - Automation
  - Workload Management
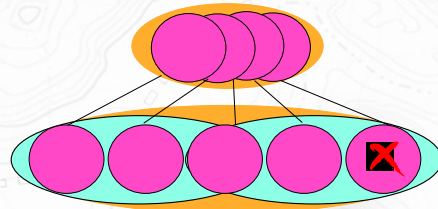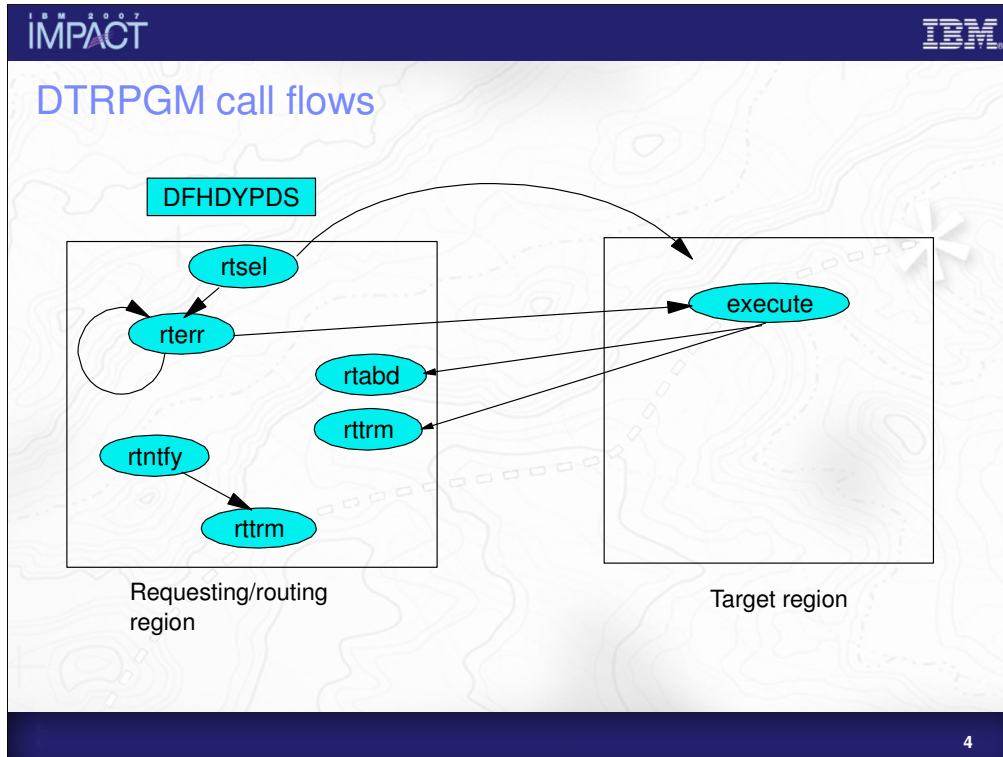 - System Management Integration
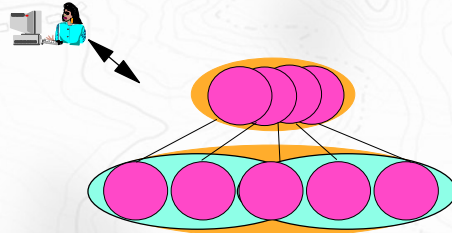
Terminology

- **Workload management**
  - Separation
  - Balancing
  - Affinity management

- **Roles**
  - Requestor
    - Initiates routing request
  - Router
    - Makes routing decision
  - Target
    - Processes work

DTRPGM call flows

DFHDYPDS

rtsel

rterr

rtabd

rttrm

execute

rtntfy

rttrm

Requesting/routing region

Target region

4

The DTRPGM SIT parameter specifies the dynamic routing program to be used for all requests that are NOT:

> non-terminal-related START requests
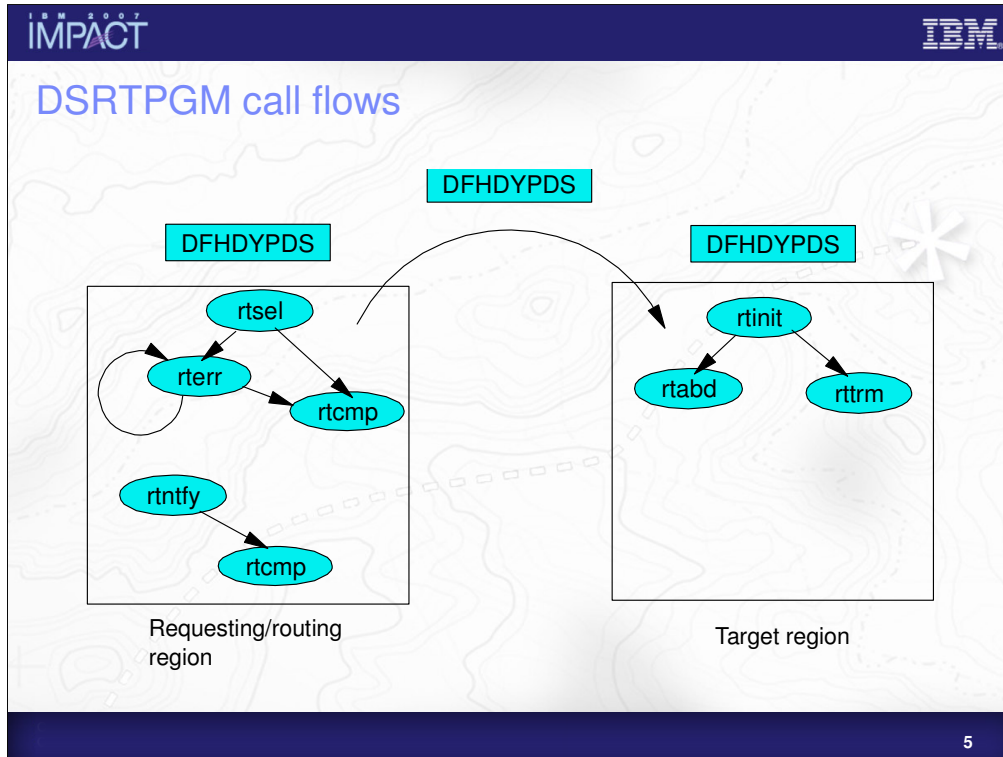>
> BTS activities
>
> Enterprise beans

CICS passes routing requests to the program via a COMMAREA named DFHDYPDS.

The routing program first selects a target region (rtsel). This may result in either the program executing in the target region or in a Route Selection Error (rterr). This could occur when links between regions are down.

If we have a Route Selection Error then we try to select another region (depending on if we need to handle affinities). This will either result in another error or an execution.

If execution abends in the target region we return to the routing region with a Route Abend (rtabd). If execution completes without abending we return with a Route Termination (rttrm).

We may be notified of a transaction being statically routed (rtntfy). In this case we do nothing and simply call route termination (rttrm).

DSRTPGM call flows

The DSRTPGM SIT parameter specifies the dynamic routing program to be used for all requests that ARE:

> non-terminal-related START requests

> BTS activities

> Enterprise beans

CICS passes routing requests to the program via a COMMAREA named DFHDYPDS.

The routing region performs a Route Selection as before. If this fails then, like DTRPGM, the Route Error routine may be called. As before this routine may try and select another region.

If a target region is successfully selected the Route Complete (rtcmp) routine is called.

Meanwhile the work is routed to the target region. The COMMAREA is passed on the routing request and the target region performs an rinit call which can result in the abend or terminate. As the COMMAREA is passed through to the target these calls can be performed in the target region.

Affinities

- Inbound request routed

- Creates data in region

- Next route in different region

- Fails (hopefully!)

inbound
request
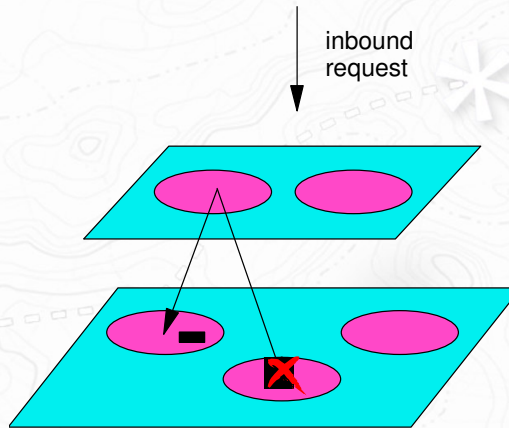
Relations and Lifetimes

- **Relations**
  - Userid – First use of user ID
  - Luname – First use of Luname
  - Global – First use of work unit
  - Bappl – Business application

Relations and Lifetimes

- **Lifetimes**
  - Delimit – Maintain until end/start transaction
  - Logon – Duration of VTAM logon
  - Signon – Duration of CICS signon
  - System – Whilst AOR is active
  - Perm – Duration of workload
  - Pconv – Duration of psuedo conversation
  - Activity – Duration of CICS BTS activity
  - Process – Duration of CICS BTS process
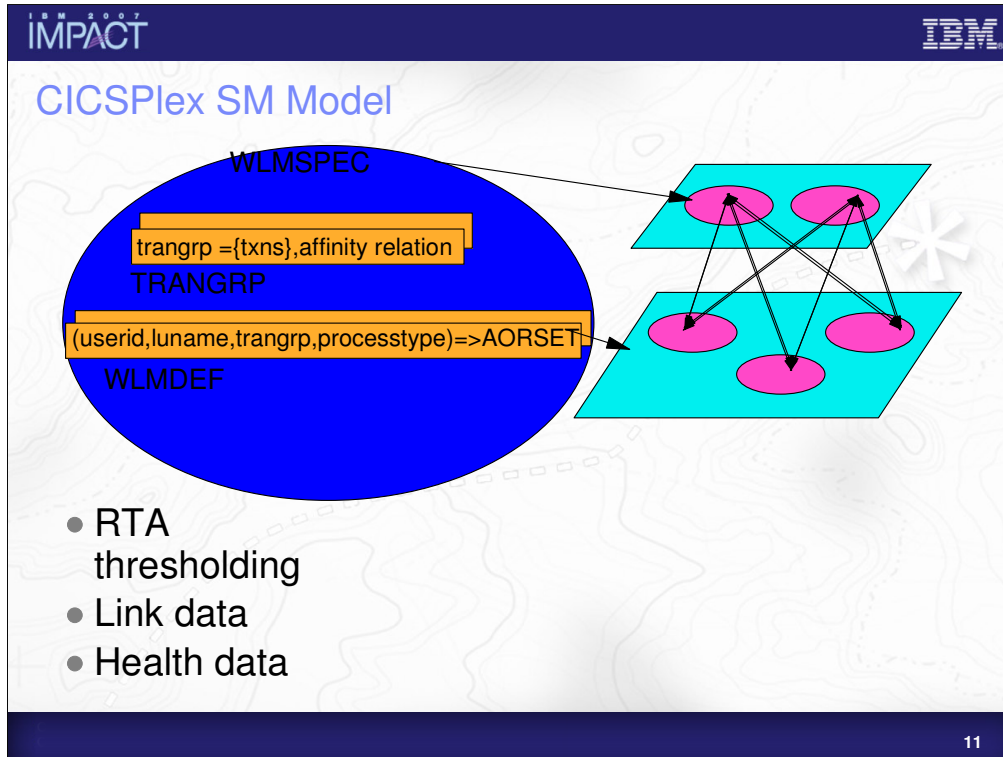
IBM 2007
IMPACT

IBM.

Affinity determination

- CICS Interdependency Analyzer for z/OS (CICS IA)

- Run-time and batch reporting tool

- Captures resource relationships such as
  - which resources (Programs, Files, TSQs TDQs etc) are required by a transaction
  - what resources are no longer used
  - the sequencing of transactions within an application
  - Includes CICS, DB2, WebSphere MQ and IMS DB resources

9

Affinity determination

- **Captures affinity relationships**
  - what programs have potential affinities due to the CICS API commands within them
  - which transactions have affinities to other transactions and the type and lifetimes of these affinities
  - which transactions have affinities to particular CICS regions

- **Relationship data loaded onto a DB2 data base**

- **Ad hoc analysis**

- **Loadlib Scanner and VSAM File Reporter**

CICSPlex SM Model

WLMSPEC

trangrp ={txns},affinity relation

TRANGRP

(userid,luname,trangrp,processtype)=>AORSET

WLMDEF

- RTA thresholding
- Link data
- Health data

11

Workload Management Specifications (WLMSPEC) – These allow you to specify the characteristics of a workload, such as: the workload algorithm, default affinity criteria and default target scopes for the workload. A WLMSPEC is associated with the routing regions for the workload.

Transaction Groups (TRANGRP) – These allow you to specify a set of transactions and any affinity information that should be associated with them.

Workload Management Definitions (WLMDEF) – In association with TRANGRPs these define workload separation and affinity requirements for a group of transactions. This is the finest level of control, i.e. if a transaction is covered by the WLMDEF the information in the WLMDEF overrides what is in the WLMSPEC. For example, if a WLMSPEC says we can route work to all regions in the CICSplex we can override that and route a transaction to only a subset of regions in the CICSplex.

11

Health Indicators

- **Basic Health indicators**
  - Maxtask
  - SOS
  - Dumping
  - Stalled

- **Region indicators**
  - Down
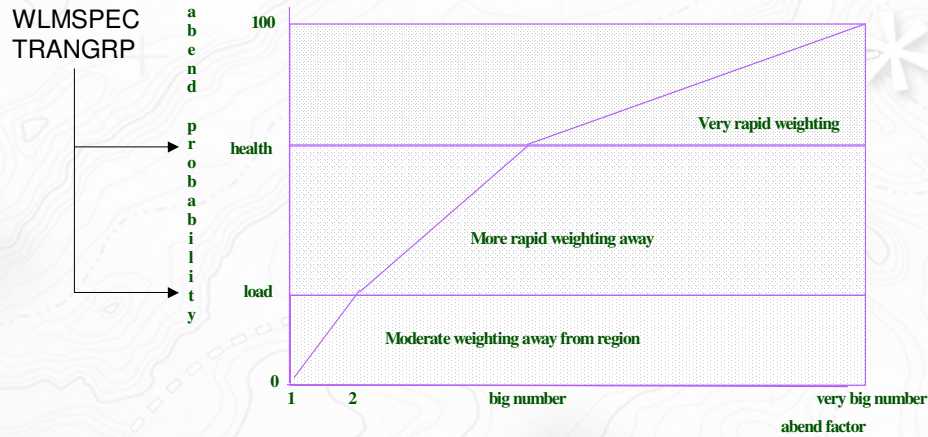 - Inactive
  - Quiescing

RTA event
  VHS/VLS
  HS/LS
  HW/LW

Link indicators
  Link down
  Lost contact
  No link

Abend health

Abend factor calculation

If the abend probability is below the load value in the WLMSPEC increases, the abend factor increases at a base rate.

If it is above the load value then it increases at a more rapid rate. If it is above the health value it increases at an even more rapid rate.

This allows you to control how sensitive WLM is to changes in abend probability.

IMPACT   IBM 2007   **IBM.**

Queue

- **Calculate weight**
  - (load*link*abend*100)+health

- **Load**
  - curtasks/maxtask

- **Link**
  - 2 sets based on % load
    - local
    - MRO
    - MRO/XCF
    - ISC

- **Tiebreaker**
  - Toss a coin

14

Queue example

| AOR1 | AOR2 | AOR3 | AOR4 |

AOR1
```
Link    : MRO/XM
Load    :55
PAbnd(ABCD):2.0
SOS     :No
Dump    : No
Stall   :No
RTA Severity :None
```

AOR2
```
Link    : MRO/XM
Load    :60
PAbnd(ABCD):6.0
SOS     :No
Dump    : No
Stall   :No
RTA Severity :None
```

AOR3
```
Link    : MRO/XM
Load    :70
PAbnd(ABCD):0.0
SOS     :No
Dump    : No
Stall   :yes
RTA Severity :None
```

AOR4
```
Link    : ISC
Load    :80
PAbnd(ABCD):0.0
SOS     :No
Dump    : No
Stall   :No
RTA Severity :None
```

All regions maxtask 100; abend load=2.0; Abend health=6.0

weight = (link*load*Abnd*100) + Health

(1.0*0.55*2.0*100) + 0=110

(1.0*0.7*1.0*100) + 1000 = 1070

(1.0*6*2000.0*100) + 0 = 140000

(1.3*0.8*1.0*100) + 0 = 104

Route

VALUES ARE FOR ILLUSTRATIVE PURPOSES ONLY

Goal Mode

- Utilizes MVS WLM
  - You must define average response time goals to MVS WLM

- MVS WLM provides
  - Service definition (1 per sysplex)
    - Service policies (1 active)
      - Workload definitions
      - Service classes
      - Report classes
      - Goals (Velocity,discretionary,responsetime(average or percentile)
      - Classification for CICS is
      - (subsystype,subsysinstance,userid,tranid,trangrp,LU)=>service class

Performance Indexes

- Performance index = curAvg/goal

- Response time exponentially faded
  - newAvg = totalRespTime/totalCount
  - curAvg = curAvg*alpha + (1-alpha)*newAvg

- Arrival rate exponentially faded
  - curRate = curRate*alpha + (1-alpha)*newRate

- Service classes eliminated if no arrivals in an interval

Goal Example

MVS1

| load:40 |

| load:70 |

| load:100 |

AOR1
AOR4
AOR7

Active Service class list

| arrivals:100   PI=2.0 |
| arrivals:100   PI=1.4 |
| arrivals:100   PI=1.1 |
| arrivals:100   PI=0.8 |

● **Dispatch priority**

● **Storage**

PB
PB
PB
PB
PB

MVS2

| load:40 |

| load:40 |

| load:40 |

| load:40 |

AOR2
AOR3
AOR5
AOR6

Performance Index = responsetime/goal

MVS WLM View
samples PBs for delay info
PBs go through init,running
suspended,notify(AOR)report(TOR)

REPORT allows MVS WLM to calculate it's own
PIs for the address space

A final word….

- **Balancing is probably a bad word**
  - Statistical in nature
  - Latency is allowed to offset performance cost
  - NOT round robin

- **Low transaction rates**
  - Serialised arrival rates
    - Basically random distribution into local AORS

Dynamic transaction routing

trangrp(x) = {tranid}
(userid,luname,trangrp,processtype)->SystemGroup
Associated with SystemGroup'
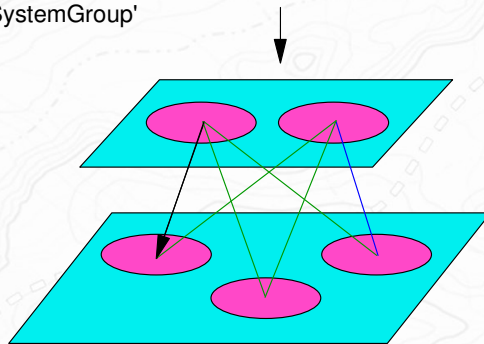
tranid(abcd)
dynamic(yes)

tranid(abcd)
dynamic(no)

Start TERMID

trangrp(x) = {tranid}
(userid,luname,trangrp,processtype)->SystemGroup
Associated with SystemGroup'

tranid(abcd)
routable(no)

EXEC CICS START
   TRANID(abcd)
   TERMID(EIBTRMID)

tranid(abcd)
routable(yes)

Start

trangrp(x) = {tranid}
(userid,luname,trangrp,processtype)->SystemGroup
Associated with SystemGroup

EXEC CICS START
TRANID(abcd)

tranid(abcd)
routable(yes)
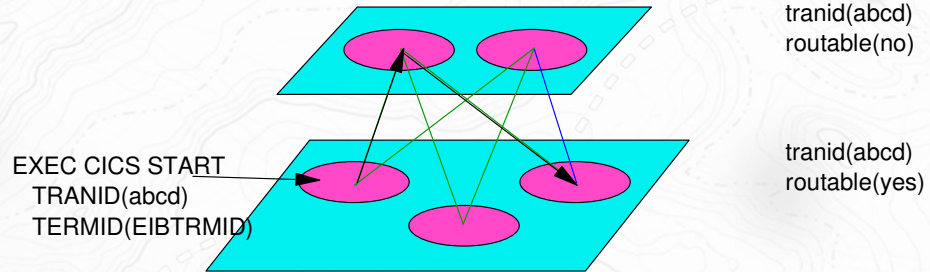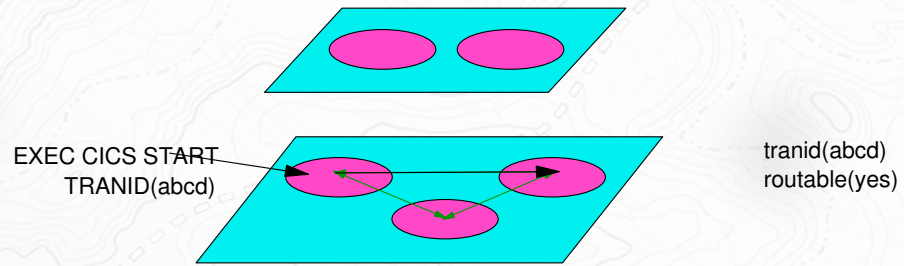
CICS BTS

trangrp(x) = {tranid}
(userid,luname,trangrp,processtype)->SystemGroup
Associated with SystemGroup'

EXEC CICS RUN
   ACTIVITY(name)
   <TRANID(abcd)>
   ASYNCH
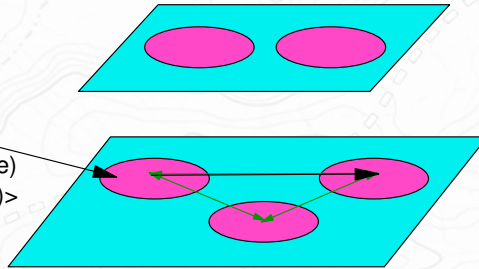
define tranid(abcd)
dynamic(yes)
routable(yes)

Dynamic DPL - inbound

trangrp(x) = {tranid}
(userid,luname,trangrp,processtype)->SystemGroup
Associated with SystemGroup'

Web, EXCI, ECI
specifying prog(xyz)
<tranid(abcd)>
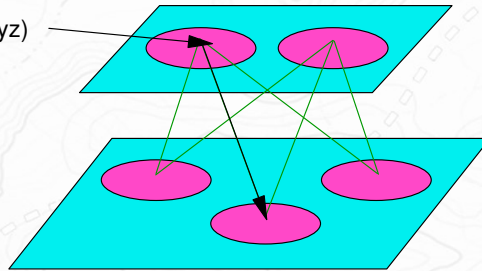
define program(xyz)
    <tranid(abcd)>
    dynamic(yes)

define program(xyz)
    dynamic(no)

define tranid(abcd)
program(dfhmirs)

Peer to Peer Dynamic DPL

trangrp(x) = {tranid}
(userid,luname,trangrp,processtype)->SystemGroup
Associated with SystemGroup'

EXEC CICS LINK
   PROGRAM(xyz)
   <TRANID(abcd)>

define program(xyz)
   dynamic(yes)

define tranid(abcd)
program(dfhmirs)

25

Dynamic WLM of EJBs

JNDI

Account
localHost

register

EJB Server

CEDA install
CorbaServer
DJAR

lookup home

h
f   Account
s

method call

localHost | listener1
| listener2
| listener3
| listener4
| listener5

DNS

routingScope

targetScope

26
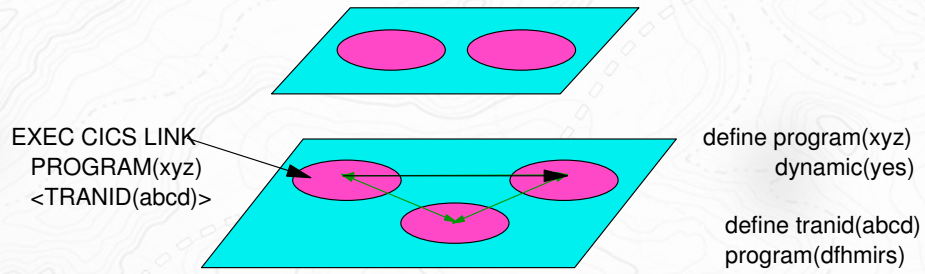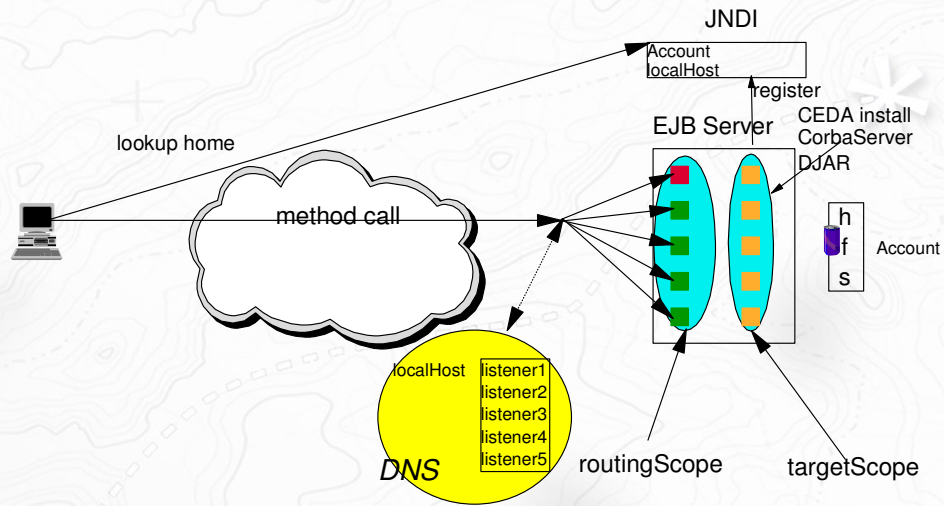
EJB target determination

DNS

Acct: TOR1
TOR2

(trangrp,lu,userid,proct)=>AORSET

Acct locate

TOR
TOR

RQMODEL
userid
beanname
methname
tranid

Client

AOR

RQMODEL
userid
beanname
methname
tranid
CorbaServer

bean

Directory

ior->AOR

27

27

Link3270 workloads

TOR

AOR

Name Server

Router
dfhbrmr

Partner
dfhbrmp

Bridge exit
dfhbrme

Client

ECI

Start brexit

User App

Client

EXCI

formatter
dfhbrmf

DPL

DPL

Client
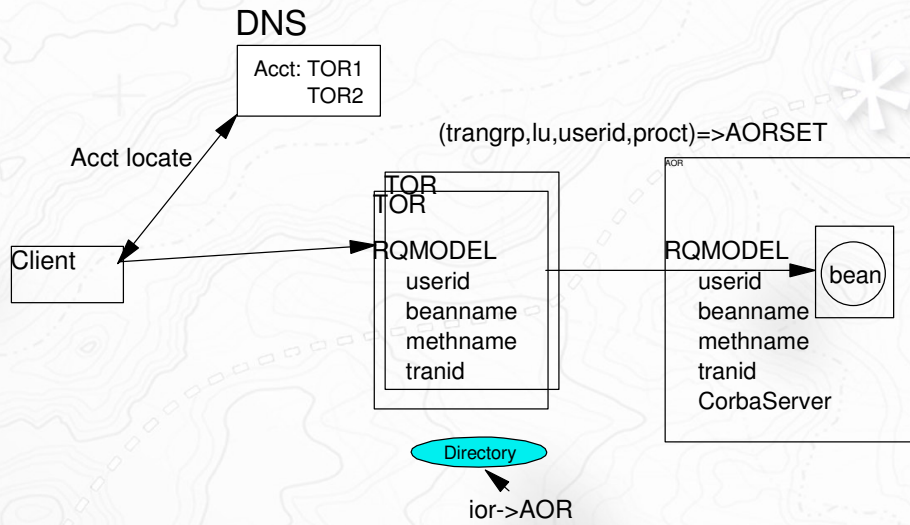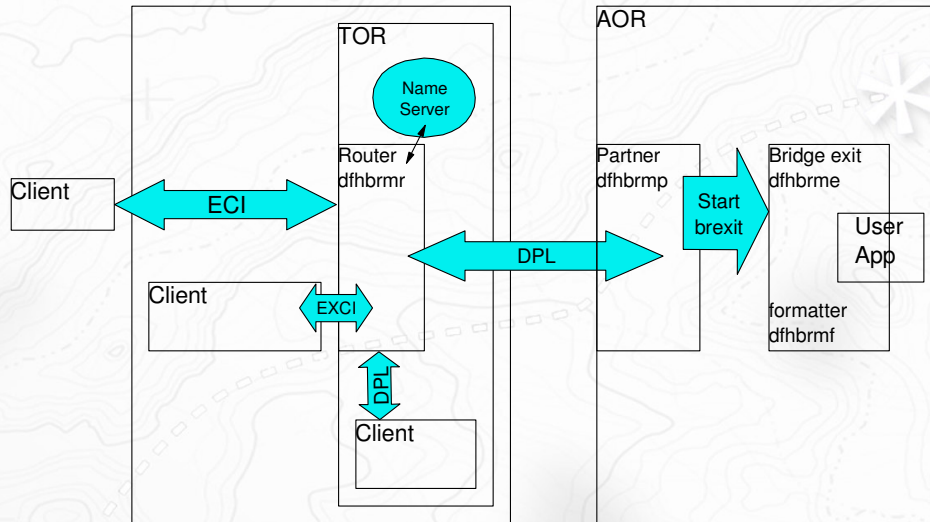
Workload separation and affinities

- **Separation allowed**
  - (userid,trangrp,luname,processtype)=>targets
    - Processtype not relevant
    - LUname only when LU predictable
      - User exit - OK
      - Normal execution – unpredictable

- **BRIDGE affinities managed by CICS code**
  - CICSPlex SM only called for
    - route select
    - route notify

29

Typical flow

```
client           CICS      DTRPGM          Target
                                           system

         ┌────────────┐
         │  fac create  ──────►│  route
         │             ◄──────┤
         │                     │  target
         │             ───────────────────────────►  application
         │             ◄───────────────────────────  execution
         │             ───────►│  rtterm
         │             ◄──────┤
      ◄──┤             │
         └───────────►│
         ┌────────────┐
         │             ───────►│  notify*
         │             ◄──────┤
         │                     │  target
         │             ───────────────────────────►  application
         │             ◄───────────────────────────  execution
         │             ───────►│  rtterm
         │             ◄──────┤
      ◄──┤  fac delete │
```

30
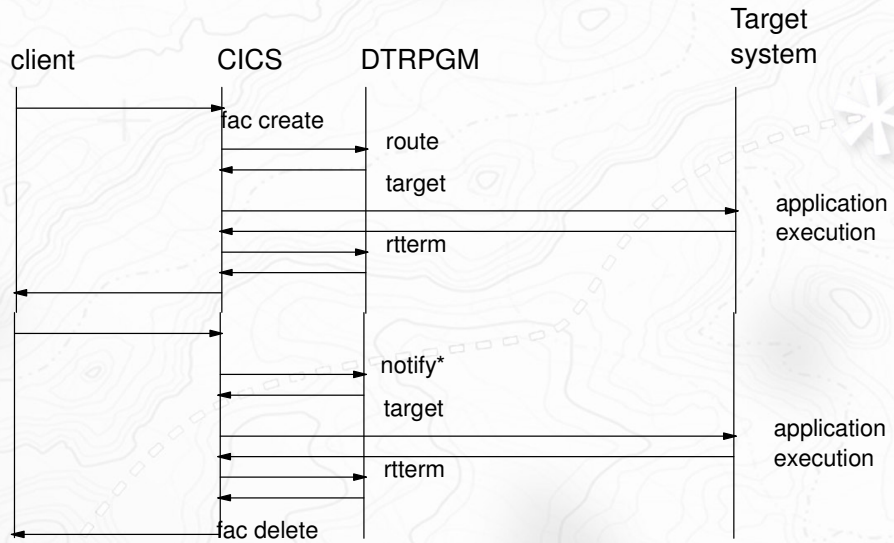
Route reject processing

- **Target regions can reject routing request**

- **Provided via user exits**
  - XICERES exit for dynamic STARTs
  - XPCERES exit for dynamic DPLs
  - If rejected, router tries another target.

- **Typical usage**
  - resource availability in region e.g. files
  - Resource status e.g. disabled transaction

31

What it can't cope with

- **Soft failures**
  - Some support via DB2
    - Failure returned to TOR who processes it as route error
    - Detectable via TIOA, COMMAREA, TCTUA then customise WRAM

- **Killer transactions**
  - If you think you have them
    - use workload separation to limit affect
    - use storage isolation if overwrite problems

Exit Customization – Why?

- **You know your transactions!**
  - Examination of TIOA
  - Examination of COMMAREA

- **Modification of TIOA data via EXEC CICS RETURN INPUTMSG**

- **Direct modification of COMMAREA**

- **Using above on invocation/return to control affinity dynamically via**
  - CREAFF
  - DELAFF

Routing data outside routing exit

- Dynamic routing information from CICS environment

- Invoked via EXEC CICS LINK PROGRAM(EYU9XOP) COMMAREA(EYURWTRA)
  - EYURWTRA contains
    - DFHDYPDS data
    - Simulated transaction info
    - Workload classification data
    - Returned applid, sysid and API fields
    - No support for Signoff/Logoff affinity
    - Will not work in goal mode

Operational control

- Can display
  - Active workloads
  - Active routing regions
  - Active target regions
    - can quiesce targets
  - Active WLM definitions
    - can discard - look for affinities first
  - Active affinities
    - can discard - be aware of consequences!
  - Transaction and TRANGRPs associated with workload

How to do it

- **Applications already exist**
  - with affinities which you don't know about
  - Lucky if you have the source
    - Luckier still if you have someone to change them to remove affinities
  - They must continue to execute during the transition to dynamic routing

36

Understand your system configuration

- **Understand your configuration**
  - Operational procedures
    - failure procedures, backup procedures

- **Design your new configuration**
  - If possible add new regions for dynamic routing
    - Easier to diagnose problems
  - If you have "spaghetti" in combined regions, move towards TOR/AOR STATIC routing first

Understand your applications

- Absolutely essential
  - Use affinities and interdependency utilities to identify relationships of parts
  - Try and move an application at a time
    - 'Baby steps' to get some success
  - Everything else being equal
    - Move shortest lifetime affinities first
      - balanced more frequently
    - Eliminate longest lifetime affinities first
      - balanced least frequently
    - Eliminate affinities that cause broadcasts
      - Reduced risk to CMAS failure
      - better performance

Getting cute on performance

- **Some affinities are worse than others**

- **Affinity Creation:**
  - Broadcast affinities
    - lock obtained across all CMASes involved in workload.
    - If any such CMAS is down,
      - lock cannot be obtained so affinity cannot be created
      - transaction instance cannot execute.
  - If affinity exists, then can execute in absence of CMAS.
    - pconv, logon, signon, delimit must be same incarnation of AOR

Getting cute on performance

Broadcast Affinity	Permanent	System	Logon	Signon	Delimit	Pconv	activity	process
Global	Yes	Yes	N/A	N/A	N/A	N/A	N/A	N/A
Luname	Yes	Yes	No	N/A	No	No	N/A	N/A
UserID	Yes	Yes	N/A	No	No	No	N/A	N/A
Bappl	Yes	Yes	N/A	N/A	N/A	N/A	Yes	Yes

40

Steps to exploitation

VTAM Generic resource balancing
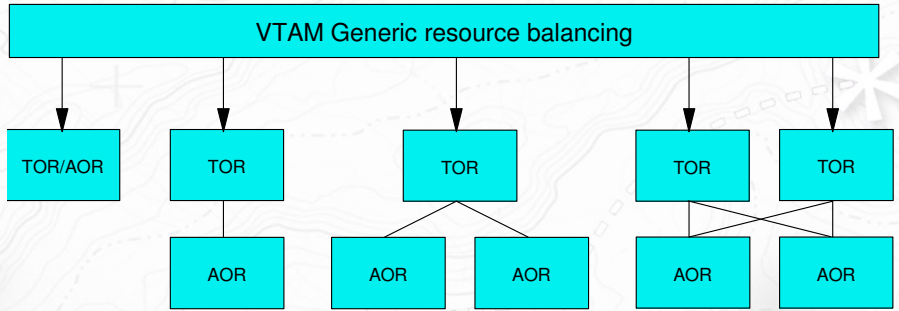
existing
system

static
routing

Dynamic routing
single TOR

Dynamic routing
multi TOR

Questions
and
Answers

43