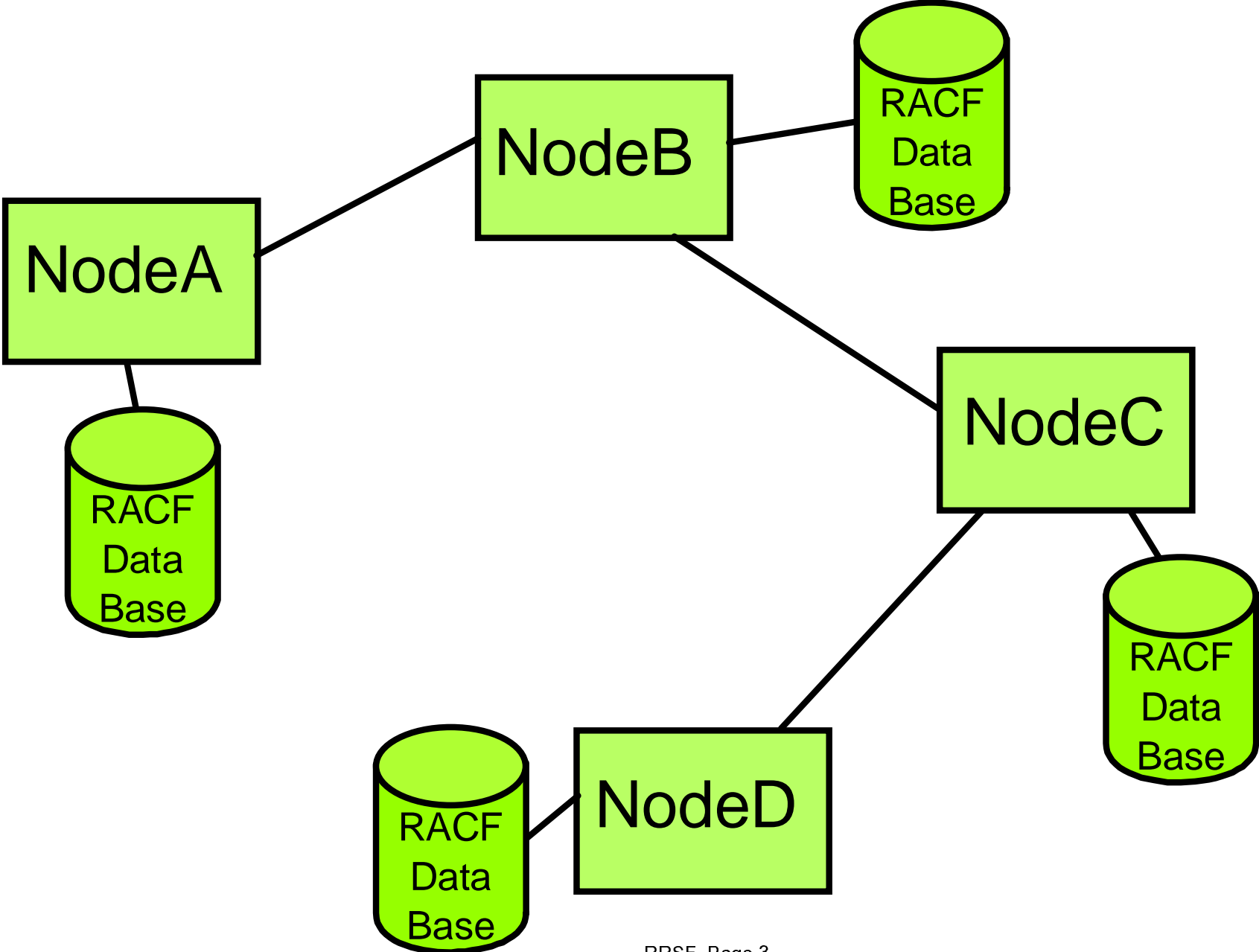


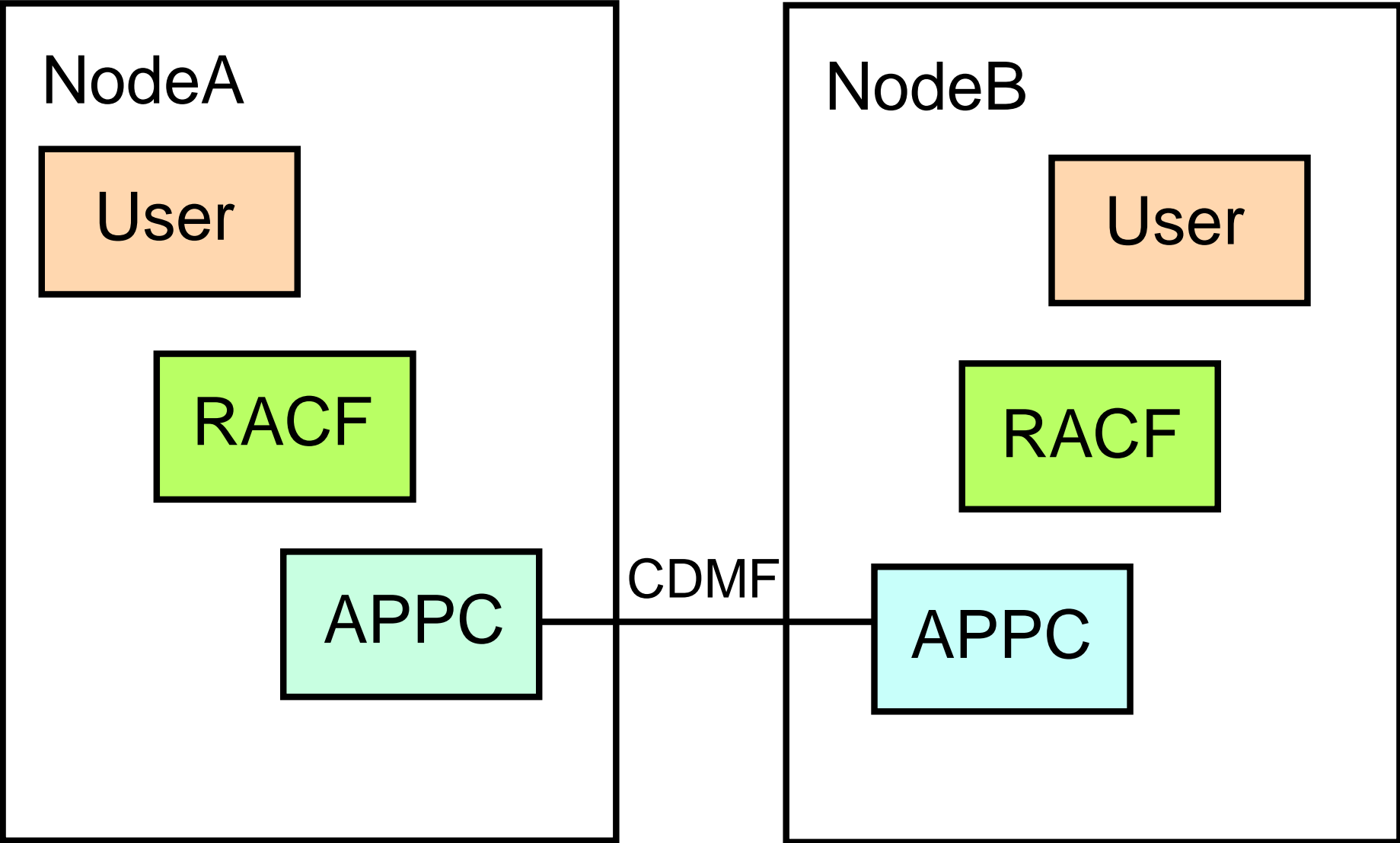
RACF Remote Sharing Facility

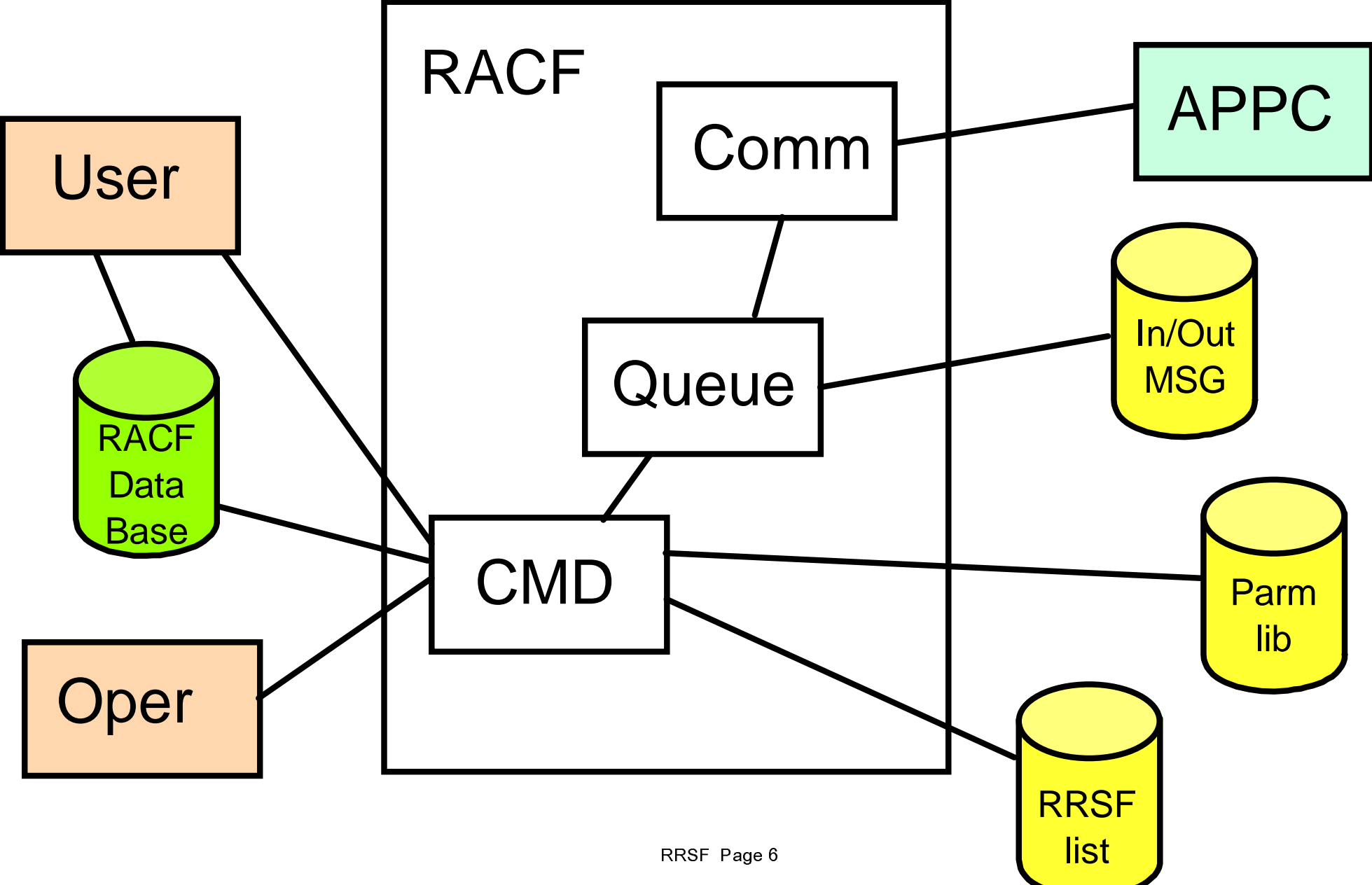
Guus Bonnes

Introduction



- Password synchronisation
 - Different userid on multiple systems
- Command Direction
 - Central administration
- Automatic password synchronisation
 - Same userid on multiple systems
- Automatic Command Direction
 - Remote sharing
- Tailorable per system/userid





- "Old" Commands
 - RVARV
 - DISPLAY
 - SIGNOFF
- "New" Commands
 - TARGET
 - SET
 - RESTART
 - STOP
- "TSO" Commands

Switch database

Persistent -

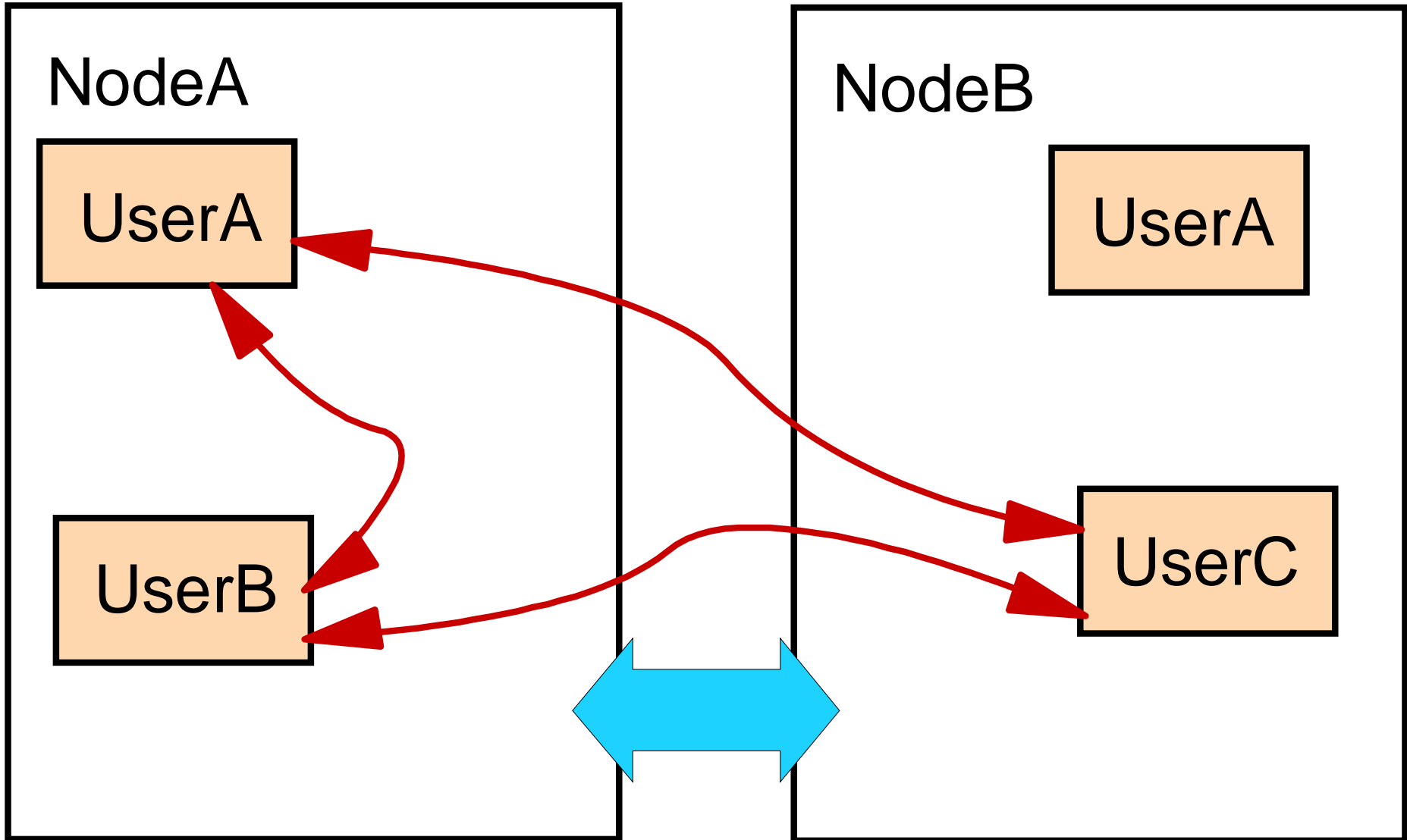
- Verification

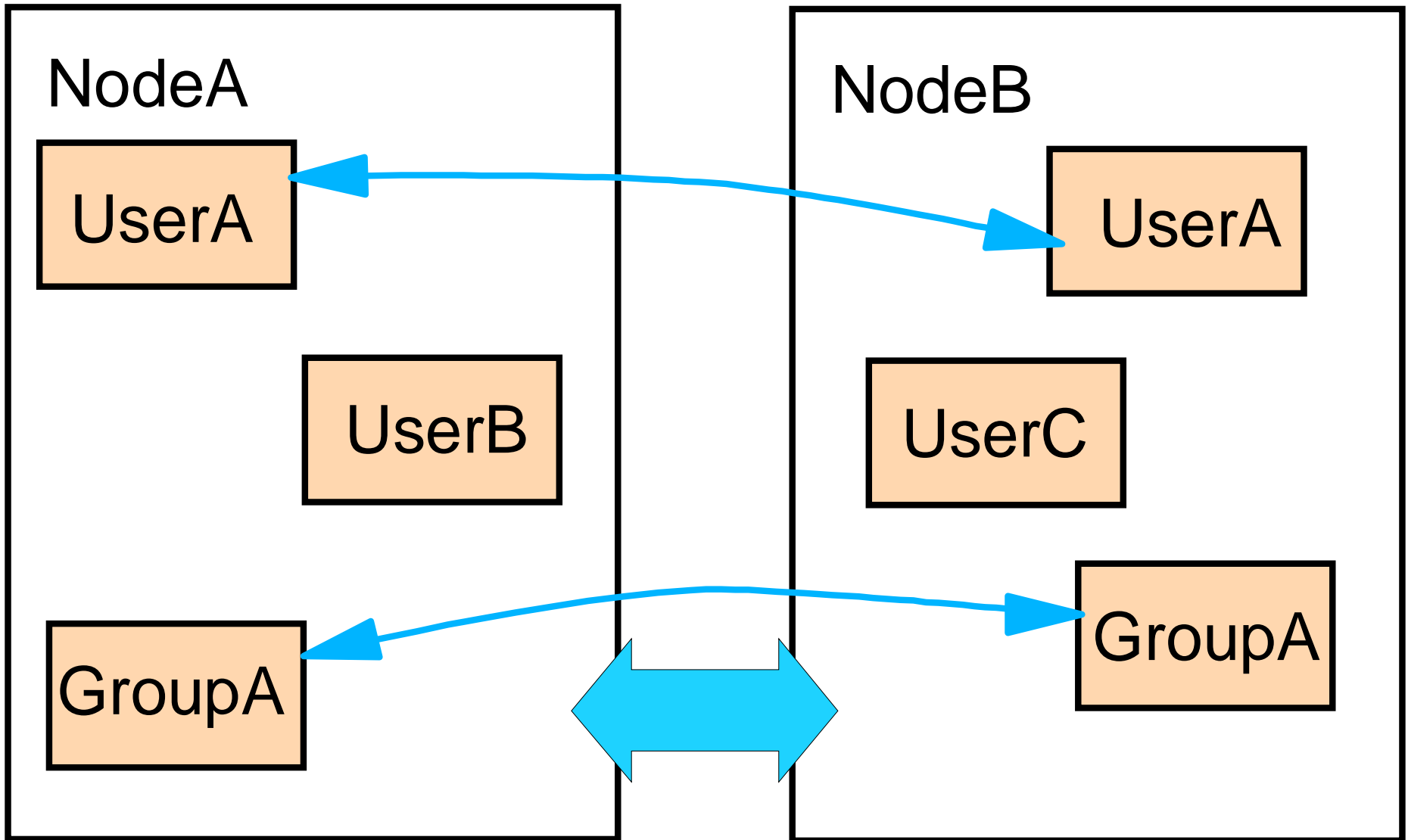
RRSF Definition

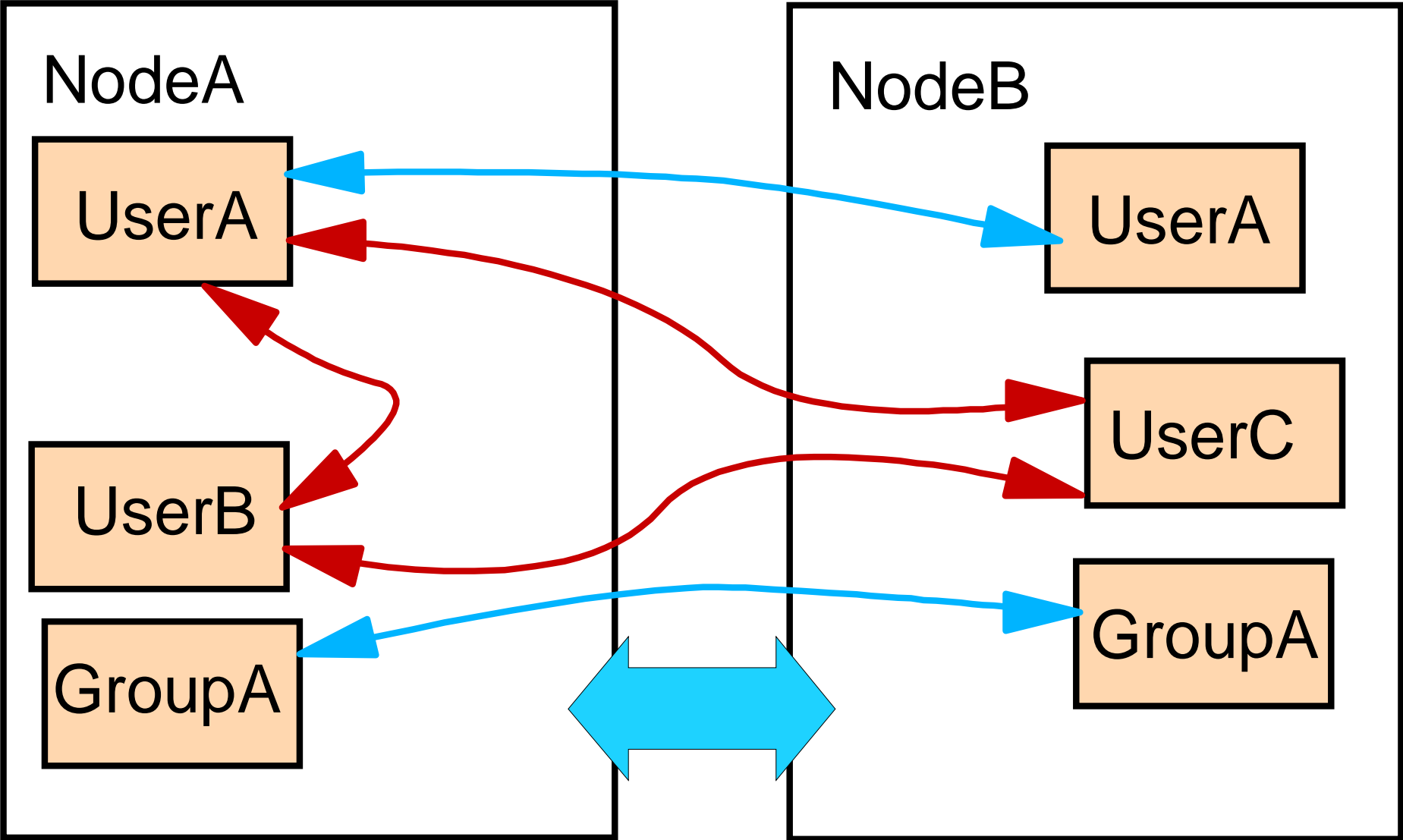
Parmlib & Options

Recovery

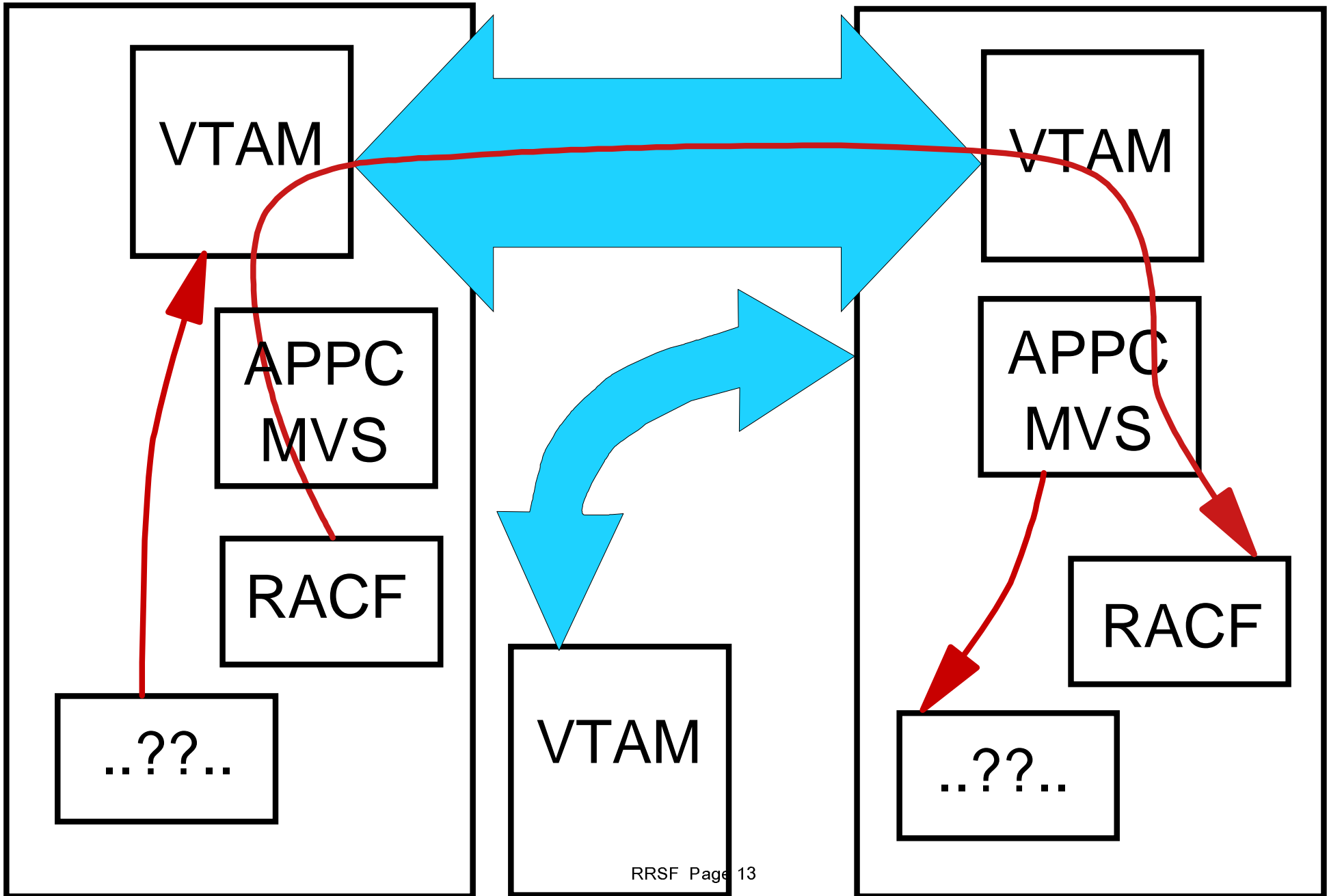
- TARGET
- SET options
- SET Include
- Allocate & IRRDPI00







Security



- Strongly recommended
 - APPCLU VTAM Partner Verification
 - VTAMAPPL Protect local VTAM ACB
 - APPCTP Protect use of Transaction Program
 - APPCSERV Protect use of TP-Server
- Recommended
 - APPCPORT Protect remote ACB Locally
 - FACILITY Protect DBTOKEN
- Optional
 - APPCSI APPC Side Information
 - APPL Use of application

- *netid.LU1.LU2*
 - *netid*
name from ATCSTR in VTAMLST
 - *LU1*
LU-name used by local RACF
 - *LU2*
LU-name used by remote RACF
- Session segment
 - *Key* Secret key for this LU-LU pair
 - *Convsec* Alreadyv is required
- VTAMLST Convsec=none !

- *ACB-name*
 - Prevent other users to pretend to be RACF
 - No users on access list needed
 - RACF address space should be TRUSTED
 - RACF runs APF authorised
 - Malicious and erroneous users

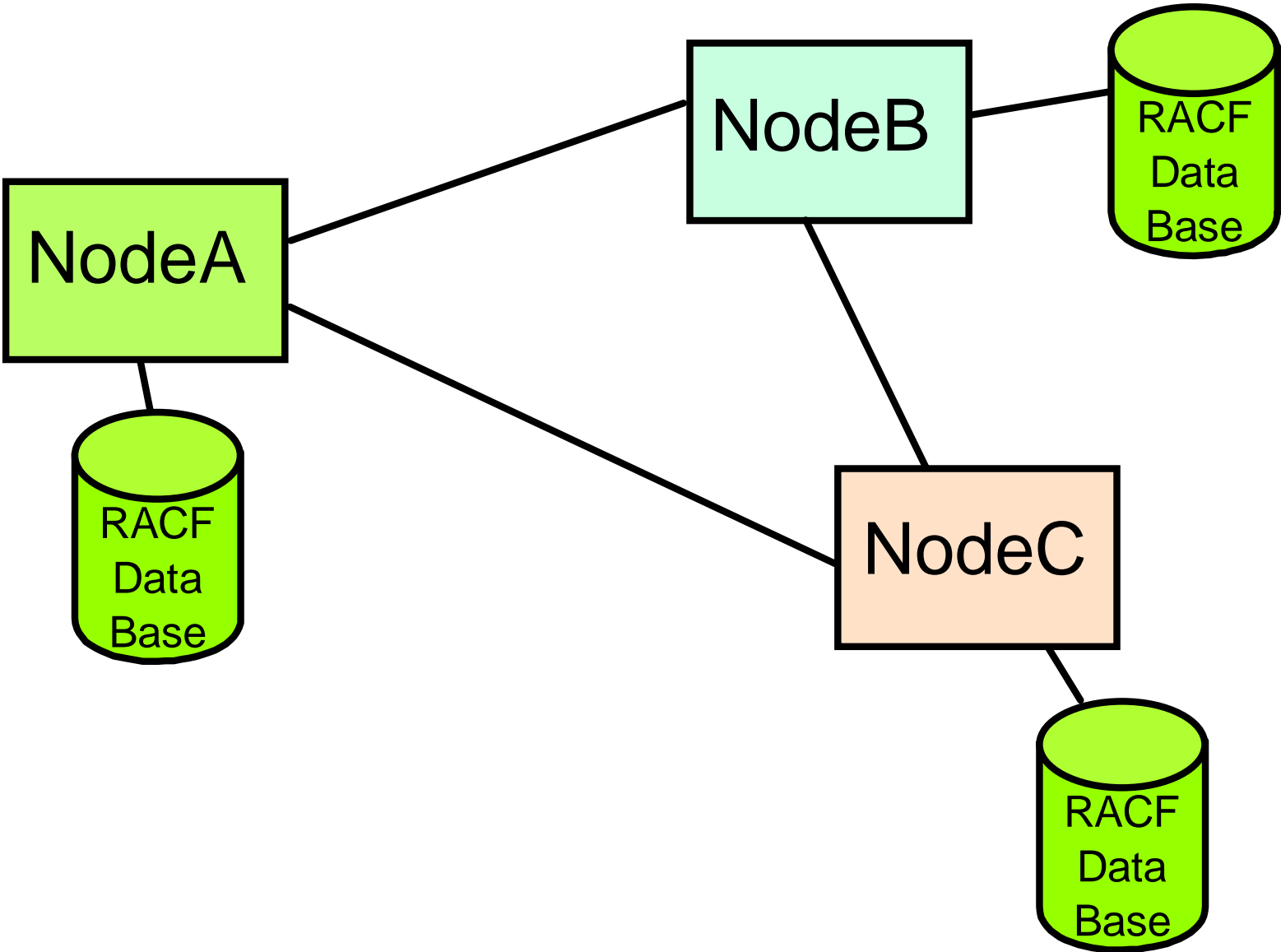
- *dbtoken.SYS1.IRRRACF*
 - *dbtoken*
Token of APPCTP database
 - *SYS1*
Scope of transaction
 - *IRRRACF*
Name of transaction
- Access list with
 - Remote RACF Address Space
 - Local APPC administrator

- *dbtoken.IRRRACF*
 - *dbtoken*
Token of APPCTP database
 - *IRRRACF*
Name of transaction
- No users on access list needed
 - RACF address space should be TRUSTED
- RACF registers itself as server for IRRRACF
 - Enhanced performance
 - No TP profile needed

- *Remote-LU-Name*
 - *Remote-LU-Name*
Name of LU used by remote RACF
- Access list with
 - Userid of remote RACF address space
- Locally protect the remote RACF system

- *APPCMVS.dbtoken*
 - *dbtoken*
Token of APPCTP database
- Access list with
 - APPC administrator authorised to maintain APPC definitions
- Database token is used in APPCTP and APPCSERV profiles. If dbtoken is changed, then the profile names should be changed as well.

Definitions



- Definitions needed in ***all*** systems
 - NodeA
 - Definition of the local node (NodeA)
 - Definition of the remote nodes (NodeB and NodeC)
 - NodeB
 - Definition of the local node (NodeB)
 - Definition of the remote nodes (NodeA and NodeC)
 - NodeC
 - Definition of the local node (NodeC)
 - Definition of the remote nodes (NodeA and NodeB)

- Name of system
- Type of system (local/remote)
- Queue datasets (inmsg/outmsg)
 - Names
 - Size
 - Locations
- Protocol
 - LU-Name
 - TP-name
 - VTAM Mode-name

- Two datasets for each connection
 - Input queue
 - *prefix.localname.remotename.inmsg*
 - Output queue
 - *prefix.localname.remotename.outmsg*
- Also two datasets for local node itself

Operator TARGET Command

BCSC

TARGET

```
[ DELETE | DORMANT | OPERATIVE ]
[ DESCRIPTION('description') ]
[ LIST ] [ LOCAL ] [ MAIN ]
[ NODE(nodename|*) ]
[ PREFIX(qualifier...) ]
[ PROTOCOL(APPC(LUNAME(lu-name)
  [ TPNAME(profile-name) ]
  [ MODENAME(mode-name) ] ))) ]
[ PURGE(INMSG | OUTMSG) ]
[ SYSNAME(sysname | *) ]
[ WDSQUAL(qualifier) ]
[ WORKSPACE(
  { [ STORCLASS(class-name) ]
    [ DATACLAS(class-name) ]
    [ MGMTCLAS(class-name) ]
  | [ VOLUME(volume-serial) ] }
[ FILESIZE([nnnnnnnnnn|500]) ] )]
```

Operator TARGET Command (example)

BCSC

```
TARGET NODE(NodeA) PREFIX(racf) -  
  WORKSPACE(VOLUME(racfxx) -  
  PROTOCOL(APPC(LUNAME(racflua) -  
            TPNAME(irrracf) -  
            MODENAME(irrmode))) -  
  DESCRIPTION('Local node NODEA') -  
  LOCAL MAIN SYSNAME(TSO1) -  
  OPERATIVE
```

```
TARGET NODE(NodeB) PREFIX(racf) -  
  WORKSPACE(VOLUME(racfxx) -  
  PROTOCOL(APPC(LUNAME(racflub) -  
            TPNAME(irrracf) -  
            MODENAME(irrmode))) -  
  DESCRIPTION('Remote node NODEB') -  
  MAIN SYSNAME(TSO2) -  
  OPERATIVE
```

● TARGET LIST

IRRM010I (#) RACF SUBSYSTEM PROPERTIES OF LOCAL RRSF NODE IDFX:

```
STATE          - OPERATIVE ACTIVE
DESCRIPTION    - "OUR OWN NODE"
PROTOCOL      - APPC
                LU NAME          - RRSFIDFX
                TP PROFILE NAME - IRRRACF
                MODENAME         - #INTER
TIME OF LAST TRANSMISSION TO - 09:02:39 OCT 21, 2005
TIME OF LAST TRANSMISSION FROM - 09:02:39 OCT 21, 2005
WORKSPACE FILE SPECIFICATION
  PREFIX          - "SYS1"
  WDSQUAL        - <NOT SPECIFIED>
  FILESIZE       - 500
  VOLUME         - Z6SYS1
  FILE USAGE
    "SYS1.ADCD.INMSG"
                    - CONTAINS 0 RECORD(S)
                    - OCCUPIES 1 EXTENT(S)
    "SYS1.ADCD.OUTMSG"
                    - CONTAINS 77 RECORD(S)
                    - OCCUPIES 1 EXTENT(S)
```

- Operative Pending Connection
- Operative Pending Verification
- Operative Active
- Operative in Error
- Dormant by Local Request
- Dormant by Remote Request
- Dormant by Mutual Request
- Dormant in Error
- ??? (Not Defined)

- Options for RRSF
- Parmlib members to be processed

Operator SET Command

BCSC

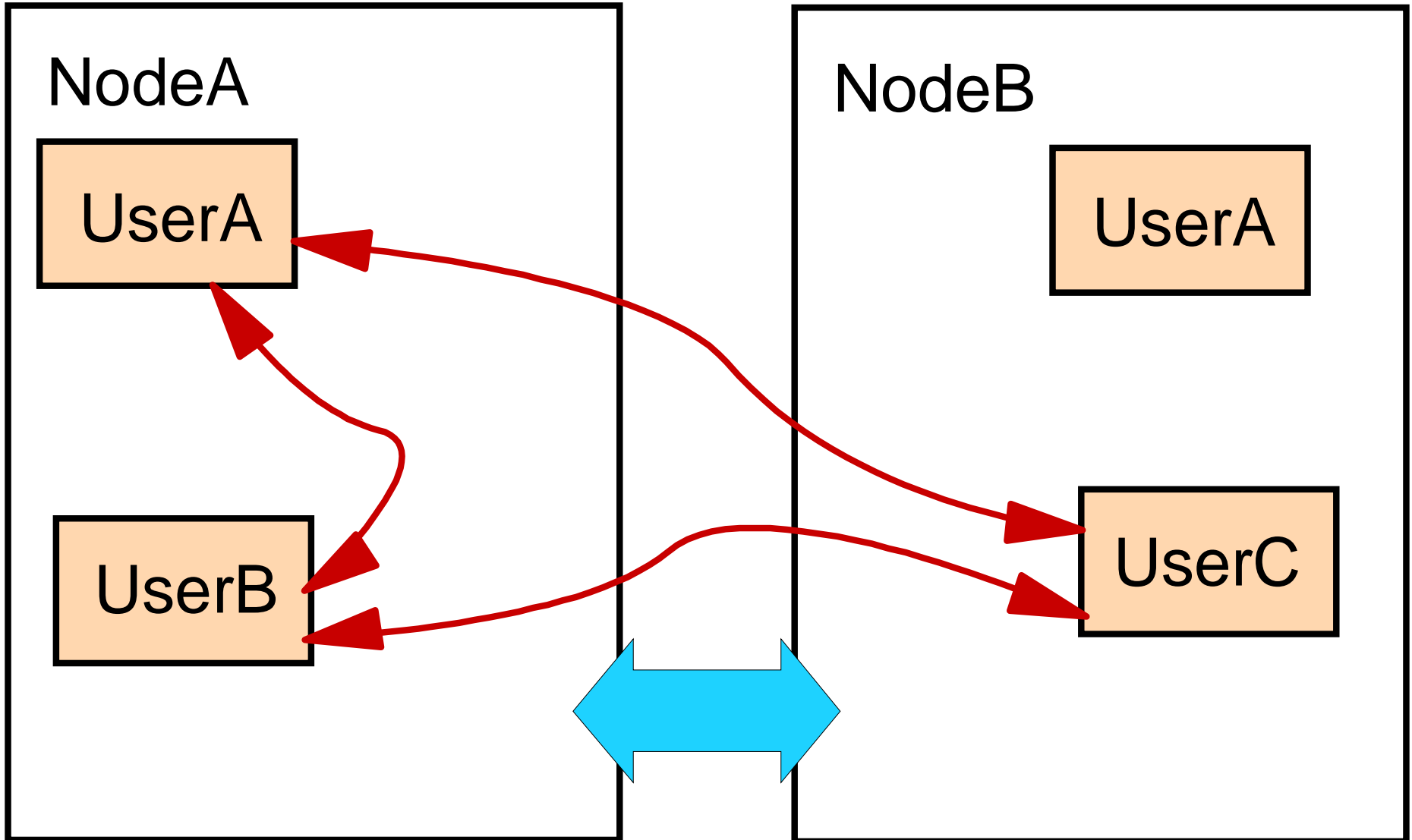
```
SET
[ auto related keywords ]

[ PWSYNCH( (
  [ NOTIFY(notify-level(list-of-notify-users))
    | NONOTIFY ]
  [ OUTPUT(output-level(list-of-output-users))
    | NOOUTPUT ]
  ) ]
  | NOPWSYNCH ]
[ INCLUDE(member-suffix...) ]
[ JESNODE(nodename) ]
[ LIST ]
[ TRACE(
  { [ APPC | NOAPPC ]
    [ IMAGE | NOIMAGE ]
    ... }
  ) ]
```

● SET LIST

```
IRRH005I (#) RACF SUBSYSTEM INFORMATION: 411
TRACE OPTIONS                - IMAGE
...
SUBSYSTEM USERID             - STCUSER
JESNODE (FOR TRANSMITS)     - N1
AUTOMATIC COMMAND DIRECTION IS ALLOWED
    OUTPUT IS *NOT* IN EFFECT
    NOTIFY IS *NOT* IN EFFECT
AUTOMATIC PASSWORD DIRECTION IS ALLOWED
    OUTPUT IS *NOT* IN EFFECT
    NOTIFY IS *NOT* IN EFFECT
PASSWORD SYNCHRONIZATION IS ALLOWED
    OUTPUT IS *NOT* IN EFFECT
    NOTIFY IS *NOT* IN EFFECT
AUTOMATIC DIRECTION OF APPLICATION UPDATES IS ALLOWED
    OUTPUT IS *NOT* IN EFFECT
    NOTIFY IS *NOT* IN EFFECT
RACF STATUS INFORMATION:
    TEMPLATE VERSION         - HRF7709 00000023.00000010
    DYNAMIC PARSE VERSION    - HRF7709
```

Userid Associations



- Three types of association
 - Peer
 - Directed commands in both directions
 - Peer with Password synchronisation
 - Directed commands in both directions
 - Password synchronisation in both directions
 - Managed
 - Directed commands in one direction

- May be initiated by either side
- Must be approved
 - Explicitly by partner
 - Explicitly by requestor (password)
 - Implicitly via administrator authority
- RACLINK command
 - Command and options may be controlled via RACF RRSFDATA profiles

RACLINK

```
[ ID(userid1...) ]  
[ LIST([node|*].[ userid2|*]...) ]  
  | DEFINE([node].userid2[/password ]....)  
    [MANAGED | PEER [(NOPWSYNC | PWSYNC )]]  
  | UNDEFINE([node].userid2...) ]  
  | APPROVE([node].userid2...) ]  
]
```

- RACLINK DEFINE(*node.user/pwd*)
PEER(*pwsync*)
- Profiles in the RRSFDATA Resource Class
 - RACLINK.DEFINE.*node*
 - RACLINK.PWSYNC.*node*
- READ access for users authorised to issue the RACLINK DEFINE command with option
- There is no profile for other functions of RACLINK

- **UserA issues**

```
RACLINK DEFINE(NodeB.UserC)
```

```
PEER(PWSYNC)
```

- ▶ **Verification for UserA on NodeA to profiles**

```
RACLINK.DEFINE.NODEB
```

```
RACLINK.PWSYNC.NODEB
```

- **UserC gets pending message**

- **UserC issues**

```
RACLINK APPROVE(NodeA.UserA)
```

- ▶ **No verification for RRSFDATA profiles**

- **UserA and UserC get approved message**

- UserS (special) issues

```
RACLINK ID(UserA) DEFINE(NodeA.UserB)
```

- Verification for UserS on NodeA to profile

```
RACLINK.DEFINE.NODEA
```

- UserA, UserB, UserS get approved message

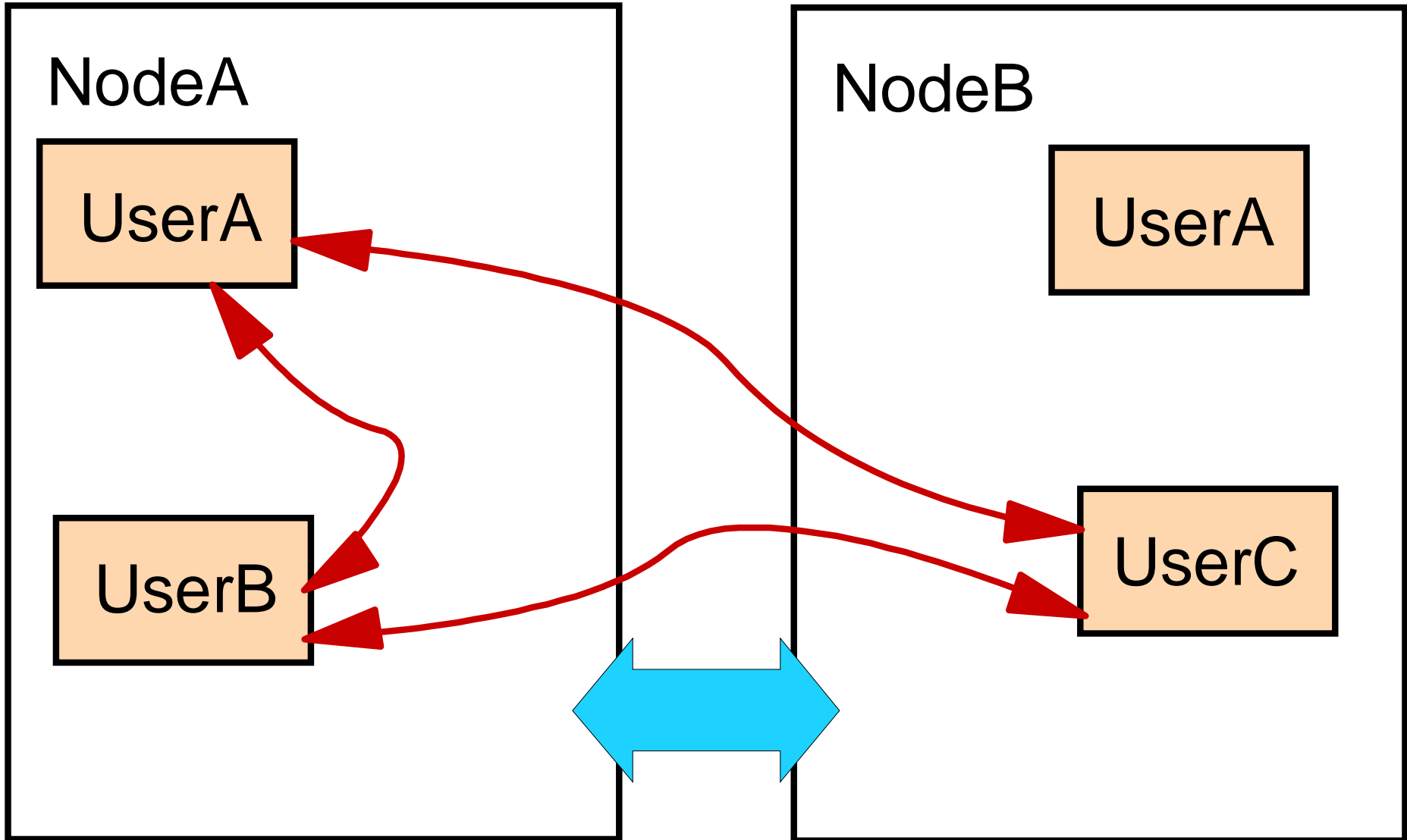
- UserB issues

```
RACLINK UNDEFINE(NodeA.UserA)
```

- ▶ No verification for RRSFDATA profiles

- UserA and UserB get deleted message

Password Synchronisation



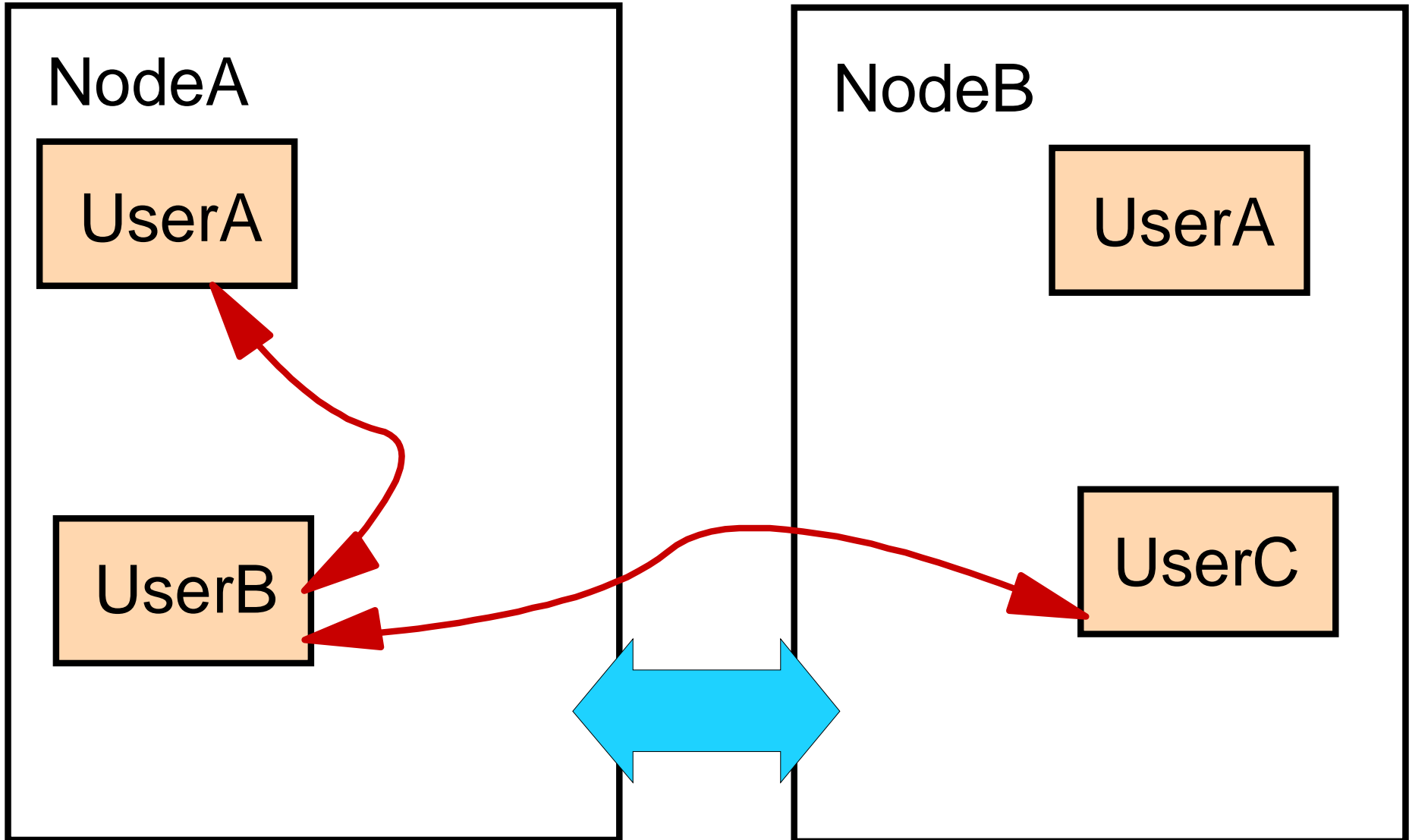
- Synchronisation when
 - Password changed via
 - Logon
 - Altuser command
 - Password command
 - RACROUTE Request=Extract (clear text only)
 - ICHEINTY (clear text only)
 - READ access to PWSYNC
 - Operator SET PWSYNCH

- To all associated users (only one level)
- Message to user and in RRSFLIST

- Profiles in the RRSFDATA Resource Class
 - PWSYNC
- READ access for people whose password may be synchronised with other userids (on the same or other system)

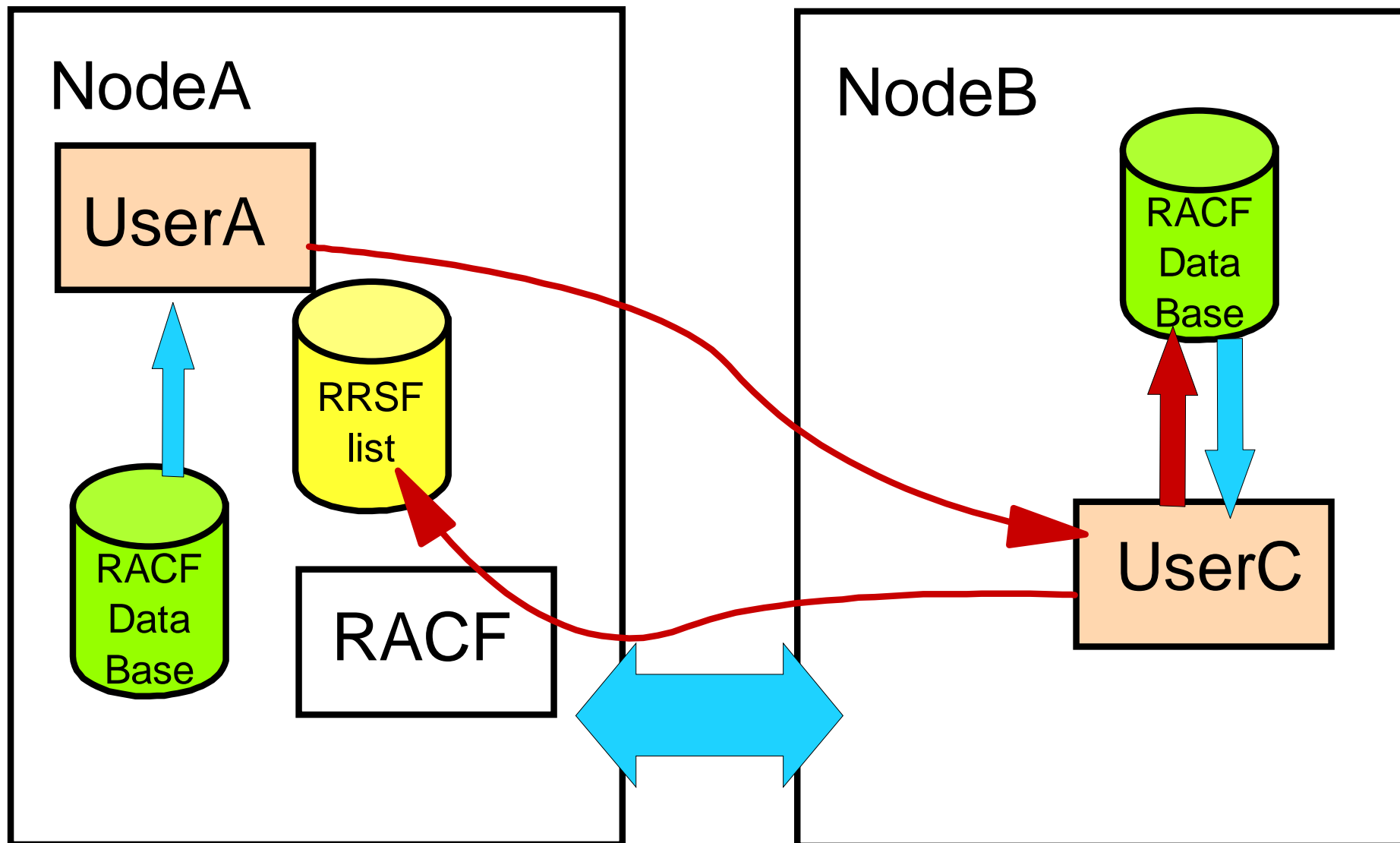
Associated Users (Incorrect)

BCSC



- To all associated users (only one level)
 - UserA changes password
 - Propagated to UserB only
 - UserB changes password
 - Propagated to UserA
 - Propagated to UserC
 - UserC changes password
 - Propagated to UserB
- Maybe best to setup centrally for end-users

Command Direction



- Additional keyword on RACF Commands
 - AT(Node.UserID)
- Command executed in RACF Address Space, Using the target userid's authority
- List of destinations possible
- Command output returned in RRSFLIST dataset
 - Suppress via "nointercom" at time of command

■ On NodeA

- UserA issues RACF Cmd
- RACF checks for approved association
- RACF checks access to *DIRECT.nodeb*



-
-
-
-
-
-

- RACF informs UserA

■ On NodeB

-
-
-
-
-
- RACF checks for approved association
- RACF builds environment for UserC
- RACF issues command

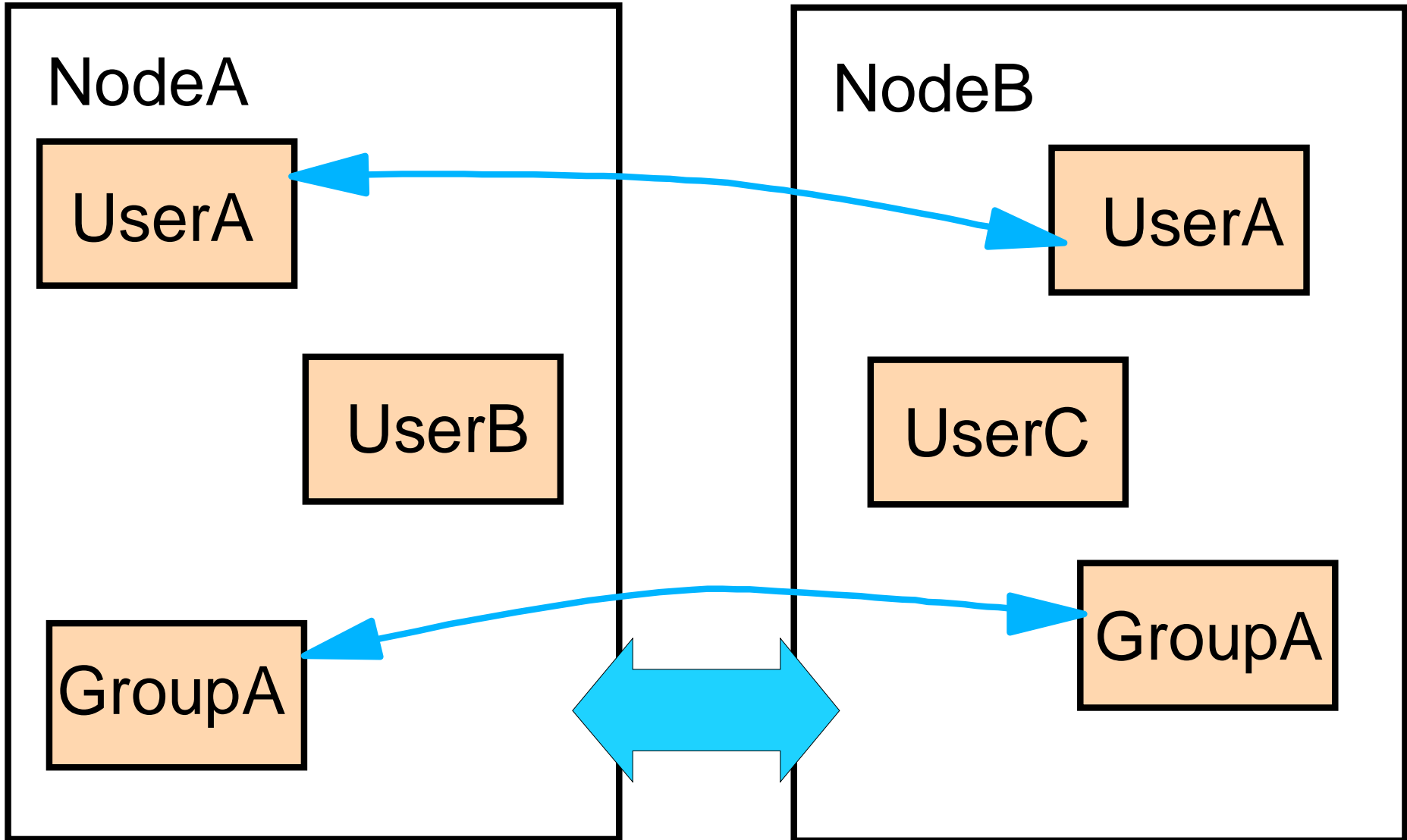


- Profiles in the RRSFDATA Resource Class
 - `DIRECT.node`
- READ access for people authorised to direct commands to node in profile name
- Userid Association is required
Exception for direction to own `node.userid`

- BLKUPD
 - RVPARY
 - RACLINK
 - RACDCERT
 - RACPRIV

 - TARGET
 - SET
 - RESTART
 - STOP
 - DISPLAY
 - SIGNOFF
- RACROUTE
 - ICHEINTY

RACF Data Base Synchronisation



- SET AUTODIRECT in source node (two places!)

```
SET
  [ AUTOAPPL([ NOTIFY(...)] [ OUTPUT(...)]) ]
  | NOAUTOAPPL ]
  [ AUTODIRECT([ NOTIFY(...)] [ OUTPUT(...)]) ]
  | NOAUTODIRECT ]
  [ AUTOPWD([ NOTIFY(...)] [ OUTPUT(...)]) ]
  | NOAUTOPWD ]

[ non-auto related keywords ]
```

level = [always | warn | fail]

user = [node.userid | &RACUID]

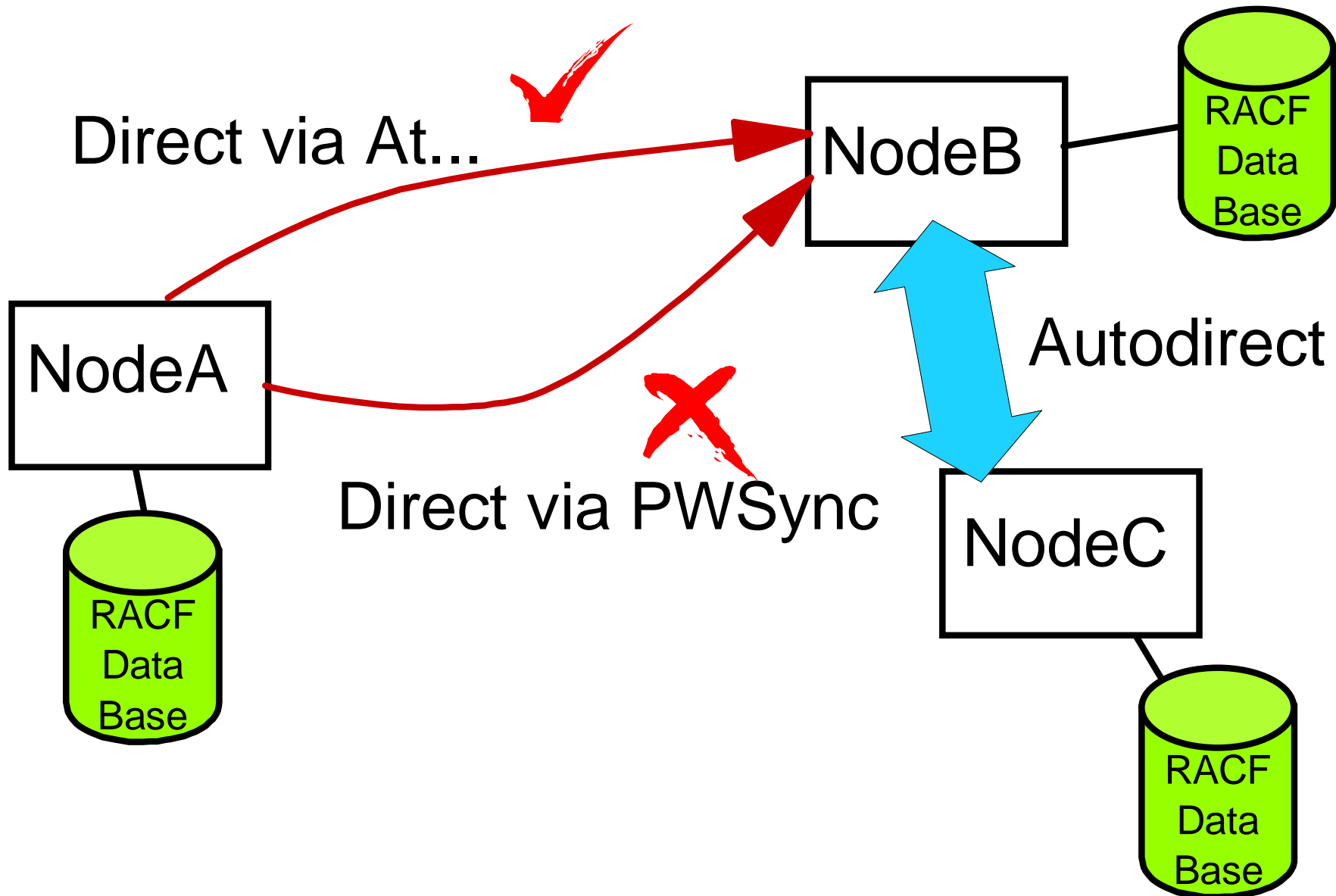
Automatic Password Synchronisation

- Synchronisation when
 - Password changed via
 - Logon
 - RACROUTE Request=Extract (clear text only)
 - ICHEINTY (clear text only)
 - Password, Altuser and Adduser password changes propagated as commands
- To all connected systems (only one level)
- Message to user

- Profiles in the RRSFDATA Resource Class
 - AUTODIRECT . *node* . USER . PWSYNC
- READ access for people whose password should be synchronised
- SET AUTODIRECT in source node (two places!)

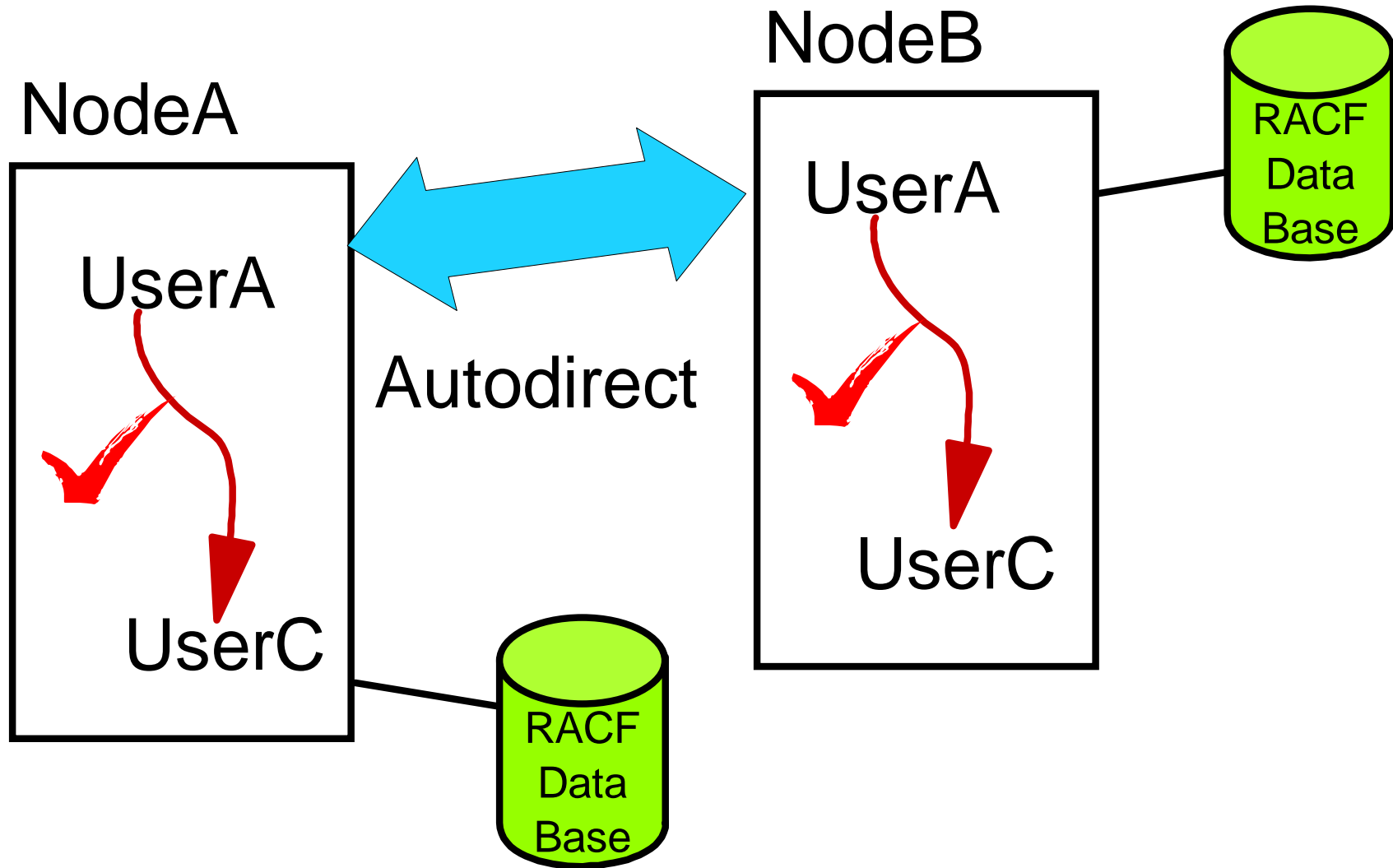
Combination with Password Synchronization

BCSC



Combination with Password Synchronization

BCSC



Automatic Command Direction

- Profiles in the RRSFDATA Resource Class
 - `AUTODIRECT .node .class .command`
- READ access for people authorised to direct commands to node in profile name
- Additional form of the profile
 - `AUTODIRECT .node .RACF .SETROPTS`
- SET AUTODIRECT in source node (two places!)

- All profiles between NodeA and NodeB
 - On NodeA
AUTODIRECT.NODEB.*.*
 - On NodeB
AUTODIRECT.NODEA.*.*
- UACC READ to synchronise independent of command issuer
 - ▶ Above samples also allow pwsync

- All profiles between NodeA and NodeB except CICS related profiles

- On NodeA

AUTODIRECT.NODEB.*.*	UACC(READ)
AUTODIRECT.NODEB.%CICS*.*	UACC(NONE)

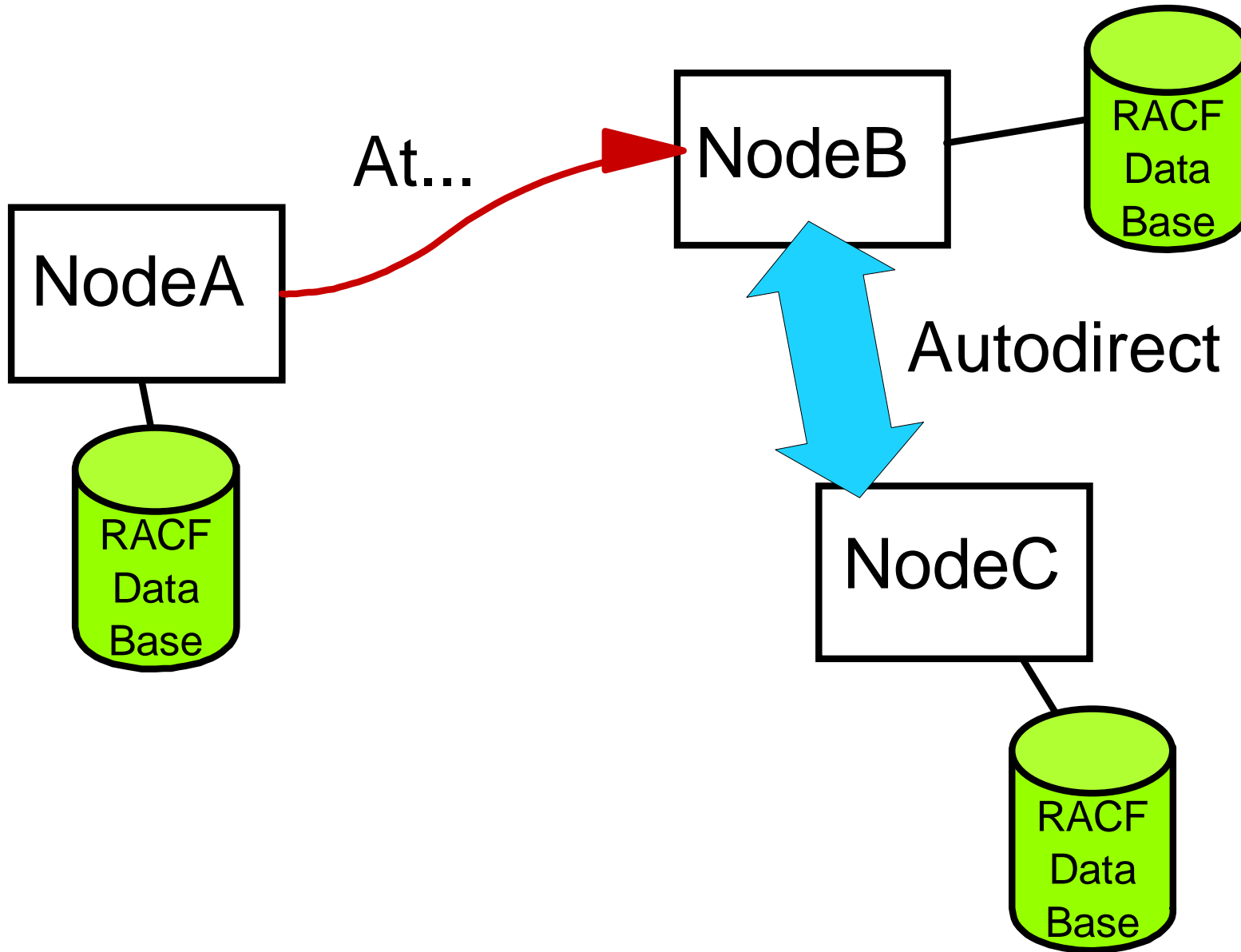
- On NodeB

AUTODIRECT.NODEA.*.*	UACC(READ)
AUTODIRECT.NODEA.%CICS*.*	UACC(NONE)

- UACC READ to synchronise independent of command issuer

- Above samples also allow pwsync

- Exceptions possible via use of
 - `ONLYAT (node.userid)`
- No automatic direction performed
- Both source and target user SPECIAL
- Command runs under specified userid
 - Same or associated userid
- Notify to command issuer
- Output to RRSFLIST



- BLKUPD
- RVARV
- RACLINK
- RACDCERT
- RACPRIV
- TARGET
- SET
- RESTART
- STOP
- DISPLAY
- SIGNOFF
- LISTUSER
- LISTGRP
- LISTDSD
- RLIST
- SEARCH
- SETR LIST
- RACROUTE
- ICHEINTY

Automatic Direction of Application Updates

BCSC

- Profiles in the RRSFDATA Resource Class
 - AUTODIRECT.node.class.APPL
- READ access for people authorised to direct commands to node in profile name
- Additional form of the profile for datasets
 - AUTODIRECT.node.DATASET.APPL (generic)
 - AUTODASD.node.DATASET.APPL (discrete)
 - AUTOTAPE.node.DATASET.APPL (discrete)
- SET AUTOAPPL in source node (two places!)

Auditing

- Use OPERCMDS resource class
- Require operators to signon
- . . .

- Parmlib commands are never audited

- Normal command auditing
- RACLINK command
 - Writes SMF at RACLINK DEFINE
 - Source system
 - Writes SMF at RACLINK APPROVE
 - Source and target system
- Access to RRSFDATA profiles
 - Success and failed access