



INTELLIGENT
ARCHITECTURE FOR
AUTOMATIC RESOURCE
ALLOCATION IN COMPUTER
CLUSTERS

Sophia Corsava and Vladimir Getov
University of Westminster, London,
U.K.

THE SITUATION TODAY

- Expensive & Difficult
- Financial Loss
- 40% of unplanned downtime due to human error
- Support/Resources not fully automated
- HUMAN EXPERTS DIFFICULT TO FIND
- IF NETWORK and/or HOSTS GO DOWN SO DO SERVICES!!!

The ideal Data Center

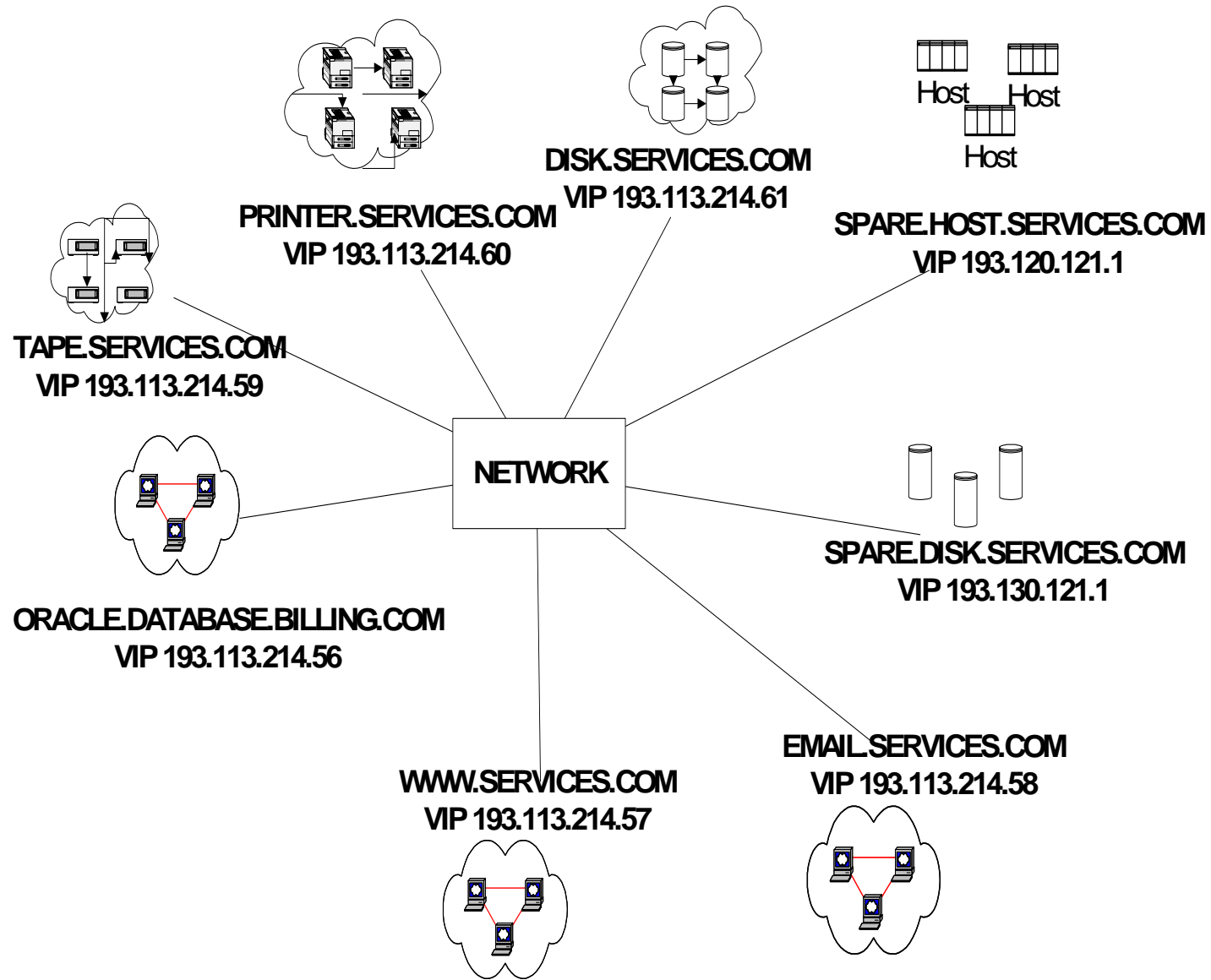
- 100% uptime, high performance reliability, availability, serviceability, flexibility, FT
- Self-sufficient & Cost effective
- Easy to maintain and support
- Detailed reporting
- Self-Managing
- Resource effective

HOW TO ACHIEVE IT



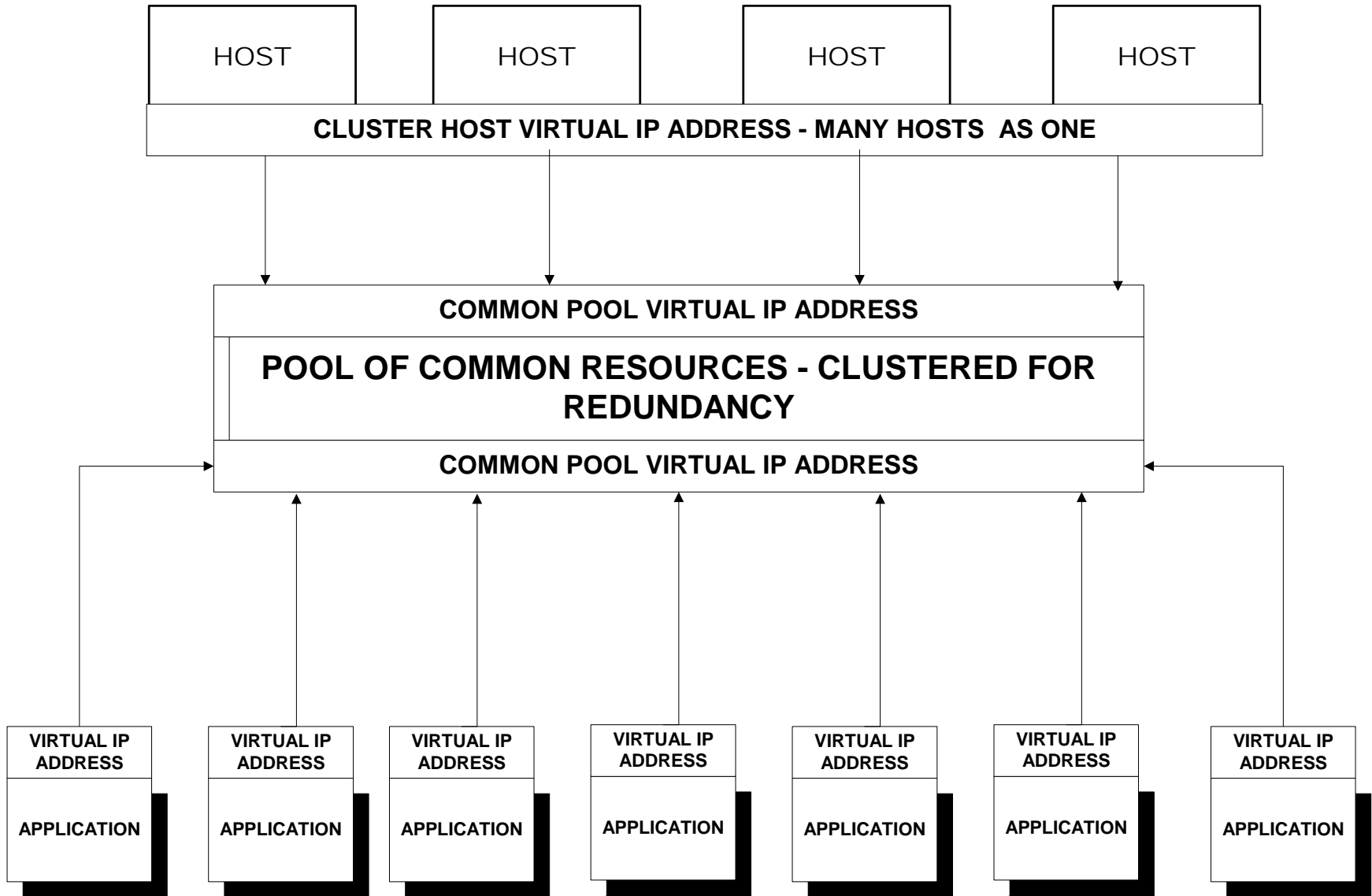
INTELLIGENT CLUSTERS

- LARGE APP CLUSTER GROUPS USING A MODIFIED COMMERCIAL CLUSTER SOFTWARE
- INTELLIGENT AGENTS/INTELLIAGENTS
- STATIC & DYNAMIC ONTOLOGIES
- FULL AUTOMATION OF ALL TASKS
- APPLICATION VIRTUAL SPACES
- ADDRESSING WITH VIP AND DNS NAMES - A FLOATING IP ADDRESS ALIASED TO PHYSICAL IP ADDRESSES
- INTELLIAGENT PRIVATE NETWORK
- NAS or SAN for commonly shared resources on site



ENGINE LOGIC

- APPLICATIONS GROUPED IN CLUSTERS
- EACH GROUP HAS CLUSTER MASTER
- CLUSTER SOFTWARE CONTROLS PARTIALLY APPLICATION AND SERVER
- INTELLIGENT AGENT CONTROLS CLUSTER SOFTWARE, SELF HEALING, RESOURCE ALLOCATION
- APPLICATIONS CAN RUN ON ANY SERVER IN THE GROUP
- ONTOLOGIES CONTAIN ALL KNOWLEDGE
- RESOURCES AVAILABLE TO ALL HOSTS – SAN/NAS
- EXTERNAL DEDICATED CHECKPOINTS (ADMIN SERVERS)
- PVN
- AUTOMATIC AND DYNAMIC



ANY HOST CAN RUN ANY APPLICATION

INTELLIGENT AGENTS

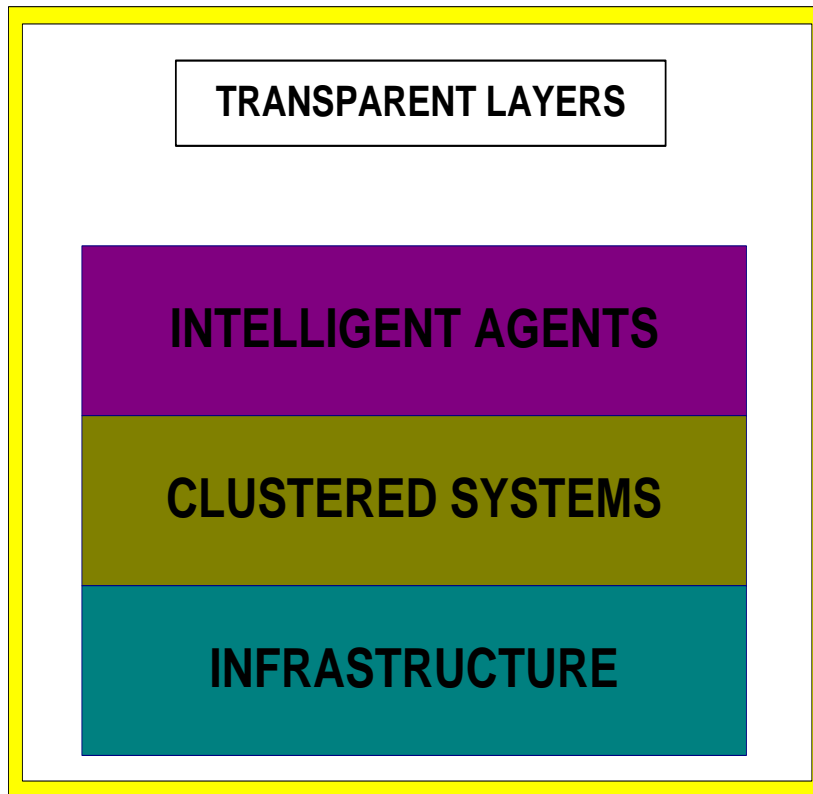


- Native OS scripts local to each host
- constraint-based causal reasoning
- ontologies
- awakened by cron
- Communicate with OS use exit codes and IPC
- Communicate with apps -try to use apps and check exit codes
- use FLAGS to indicate status and compare dynamic vs static ontologies
- Environment knowledge in ontologies – NO DBASE
- Various types of agents according to function
- External administration servers “anchor” intelliagents via the PVN



MONITOR RESOURCE
DIAGNOSE FAULT STATICALLY AND DYNAMICALLY
SELF-HEALING ACTIONS REPAIR FAULT GLOBALLY
COMMUNICATION & LOGGING
SELF-MAINTENANCE

APPLICATIONS



ABSTRACTION LEVELS FOR THE
IDEAL MODEL

Agents work in a distributed manner and intelligence emerges from their behavior and actions. They work synergistically together and can call upon each other's modules to complete a task. They are transparent to applications/services

ONTOLOGIES

- SLKT - static local knowledge templates - what the service SHOULD BE
- DLSP - dynamic local service profiles - what the service NOW is
- ISSL - shortlists of SLKTs
- DGSP - shortlists of DLSP
- agents compare SLKT and DLSP to determine fault

ONTOLOGY EXAMPLE

```
SERVER NAME          HOST1
SERVER IP ADDRESS    IP_SYBDB1
APPLICATION RUNNING   SYB_DB1
LOCAL APPLICATION PART OF EXAMP
CONFIGURATION DIRECTORY /apps/Sybase/11.9.2
DATA_DIR_1_NAME      /apps/Sybase
DATA_DIR_1_MNT       tiger:/vol/volroot/sybdir
CONFIGURATION FILE    Syb.conf
PROCESS NAME 1       sybase
PROCESS NAME 2       XXXX
MIN NUMBER OF APPL PROC RUNNING 2
MAX NUMBER OF APPL PROC RUNNING 30
MAX No FOR APPL CONNECTIONS      30
STARTUP SCRIPT        /etc/rc3.d/s99sybase
REQUIRES_1            /apps/sybase
REQUIRES_2            /opt/sybase
DEPENDS_1             WEB1
DEPENDS_2             IP_WEB1 80
DEPENDS_3             INTER1
DEPENDS_4             IP_INTER1 6721
STOP SCRIPT           /etc/rc3.d/k99sybase
```

RESOURCES

RESOURCES NEEDED:

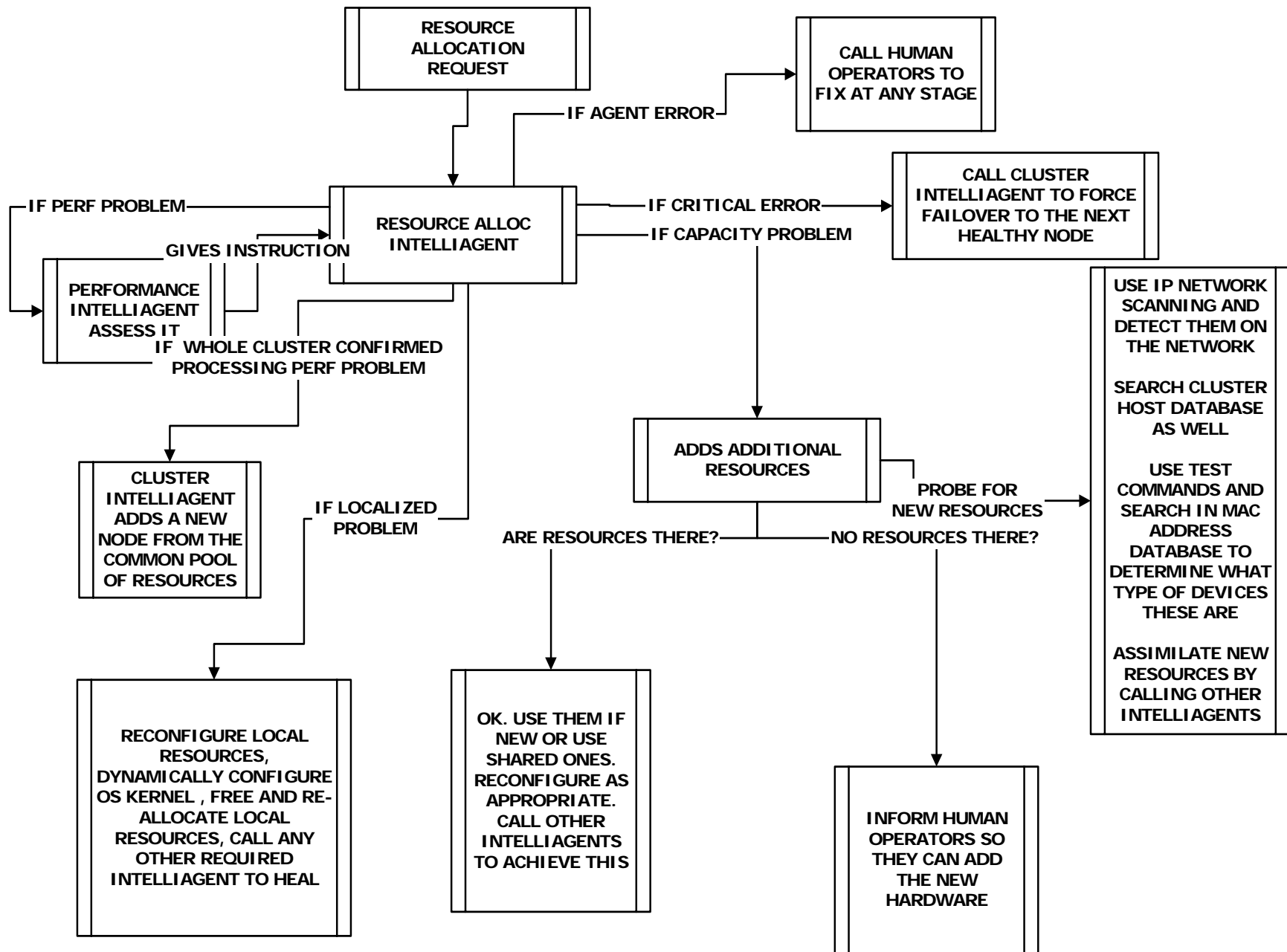
- CAPACITY
- PERFORMANCE
- FAULTS

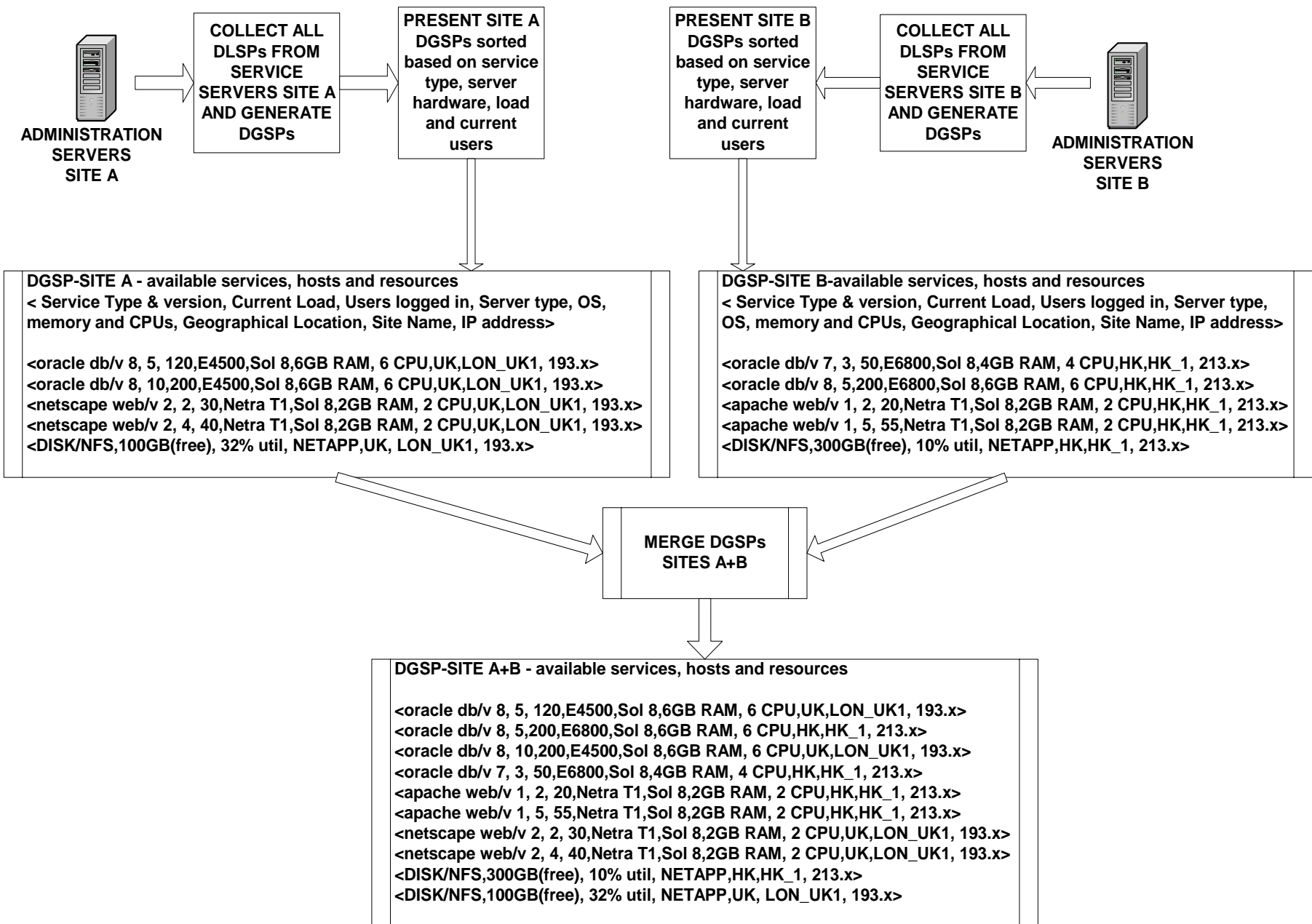


RESOURCE DISCOVERY & CONFIGURATION

A decorative graphic in the top right corner featuring a glowing globe with a network diagram overlay, showing various nodes and connections. The globe is primarily yellow and orange, with some blue and red accents. The network diagram consists of red and orange lines radiating from a central point, resembling a star or a complex network topology.

- ❖ Resources of the same type physically grouped together
- ❖ Lists of MAC addresses identify resource type
- ❖ IP scanner searches appropriate network segment if needed
- ❖ Intelliagents probe PVN every X minutes and discover new resources automatically, while they probe/monitor known ones
- ❖ Using specialized modules with admin commands, new resources identified and configured automatically – READY TO USE
- ❖ Special Resource Intelliagents, monitor, manage and heal resources
- ❖ All Resources presented within site specific lists, sorted – most avail first





OUR WORK – production site in UK ISP, E-MAIL, DBASE, WAP, GPRS, UM, IM, BILLING, E-SHOP

- GROUPED APPS IN CLUSTERS
- DEVELOPED CUSTOM AGENTS FOR EACH RESOURCE TYPE AND SERVICE
- MOVE ALL DATA ON SHARED NFS STORAGE
- DEVELOPED INTELLIAGENTS WITH BOTH FT AND SELF-HEALING ABILITIES
- AUTONOMOUS FAULT DETECTION/CORRECTION
- INSTALLED PVN

RESULTS

16 MONTHS BEFORE ANY WORK ON UK SITE

- Total Downtime = 2,166 hours
- Total Downtime from resource-related faults = 273 hours

DATA PROVIDED BY THE CUSTOMER
USING BMC PATROL

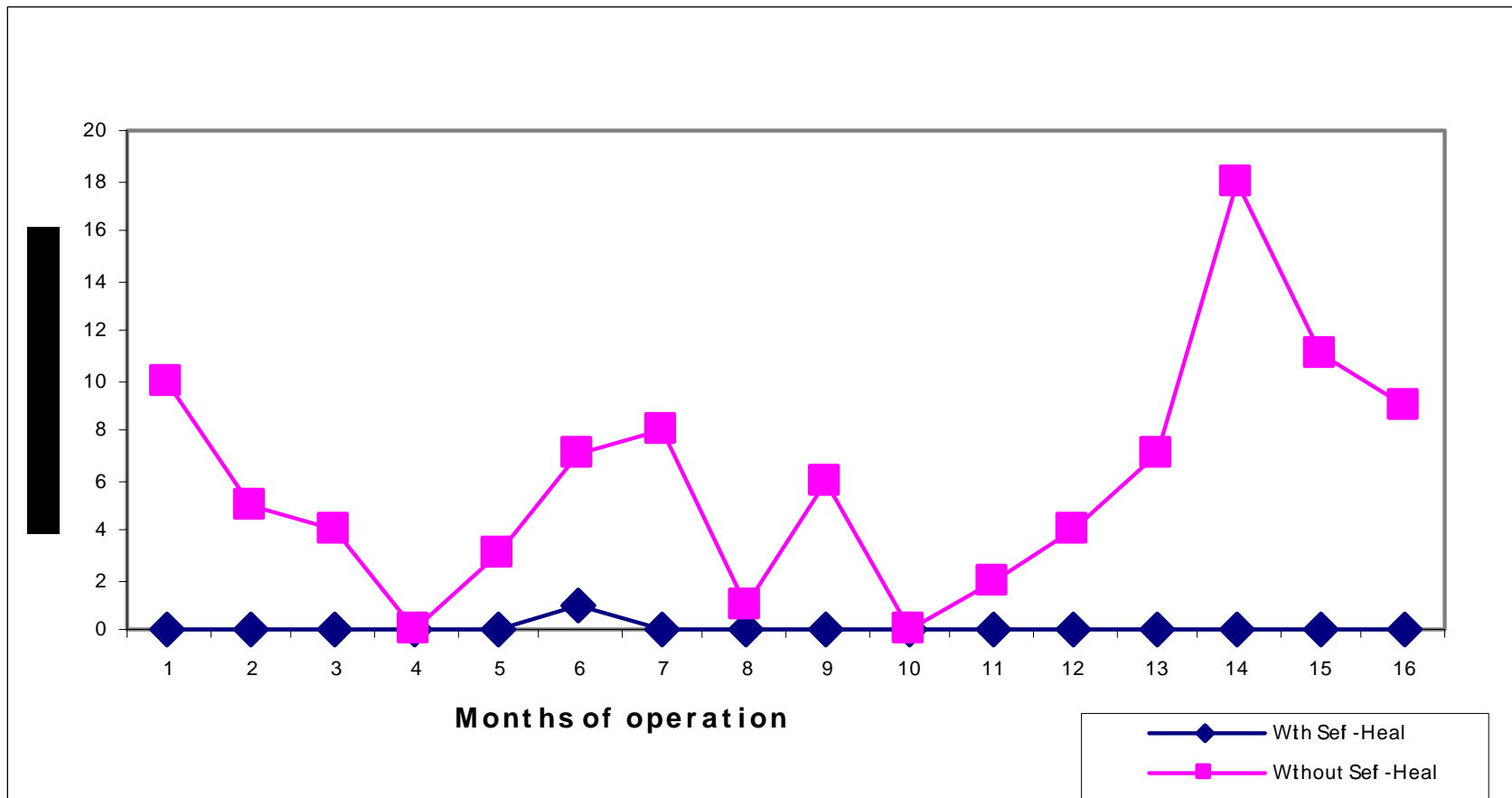
RESULTS

16 MONTHS AFTER OUR WORK

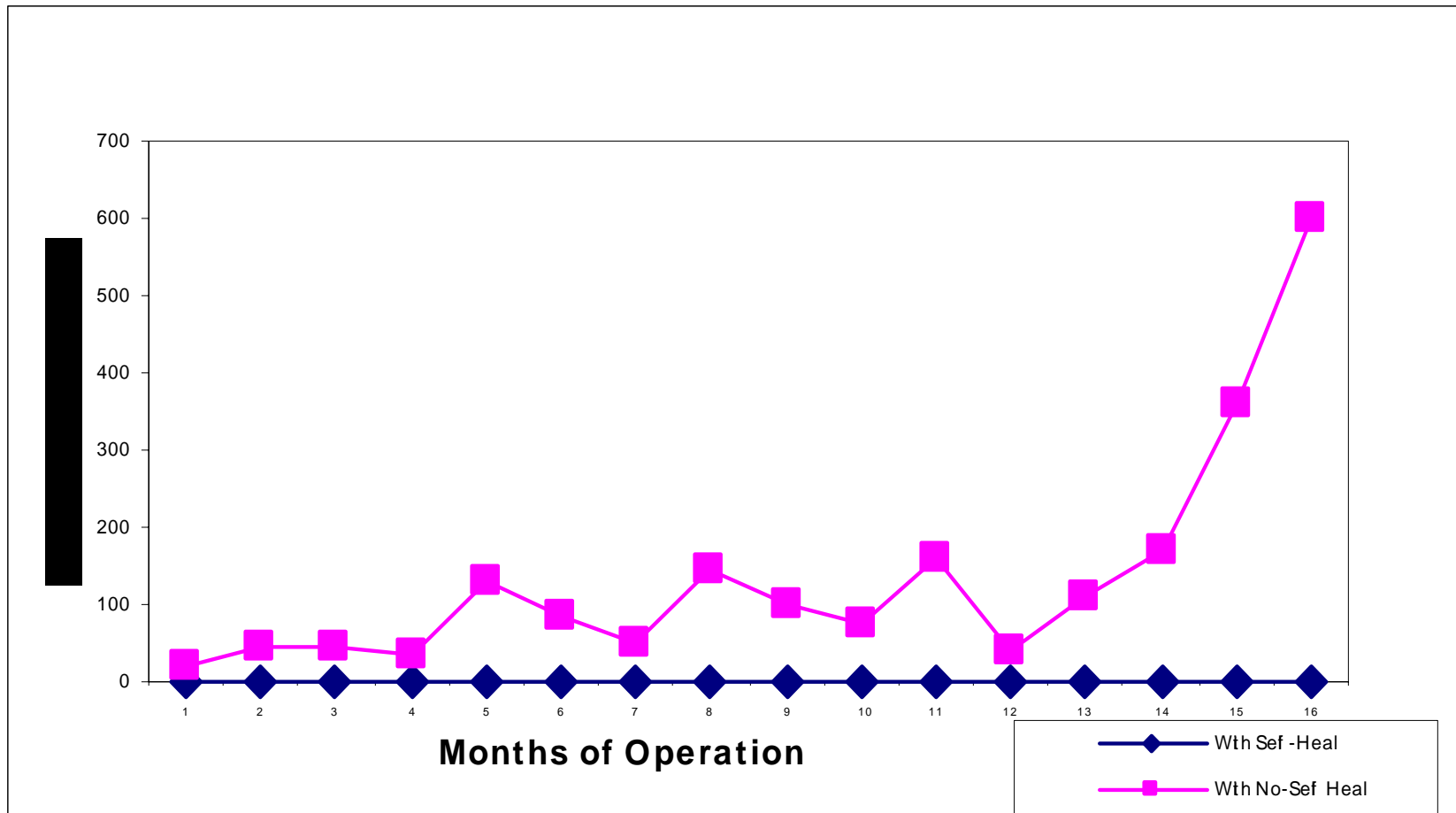
- Total Downtime = 13 hours
- Total Downtime from s/w & h/w errors = 9 hours, 3 hours of human errors due to inexperience
- ONLY ONE (1) HOUR FROM RESOURCE-RELATED FAULTS

DATA PROVIDED BY THE CUSTOMER USING BMC PATROL

Comparison of downtime from resource incidents in hours per month, 16 months before and 16 months after our work



Comparison of downtime from all types of faults in hours per month, 16 months before and 16 months after our work



Conclusions

- BECAUSE HIGH BUSINESS DEMANDS, WE NEED BOTH FT and Self-healing
- OLD CONFIGURATIONS OBSOLETE
- NEW WAY OF THINKING
- NEW ARCHITECTURES
- SELF-HEALING INTELLIGENT CLUSTERS
- BUT WORK REMAINS TO BE DONE!!!!

FURTHER INFORMATION

Sophia Corsava

SOPHIAC6@YAHOO.COM

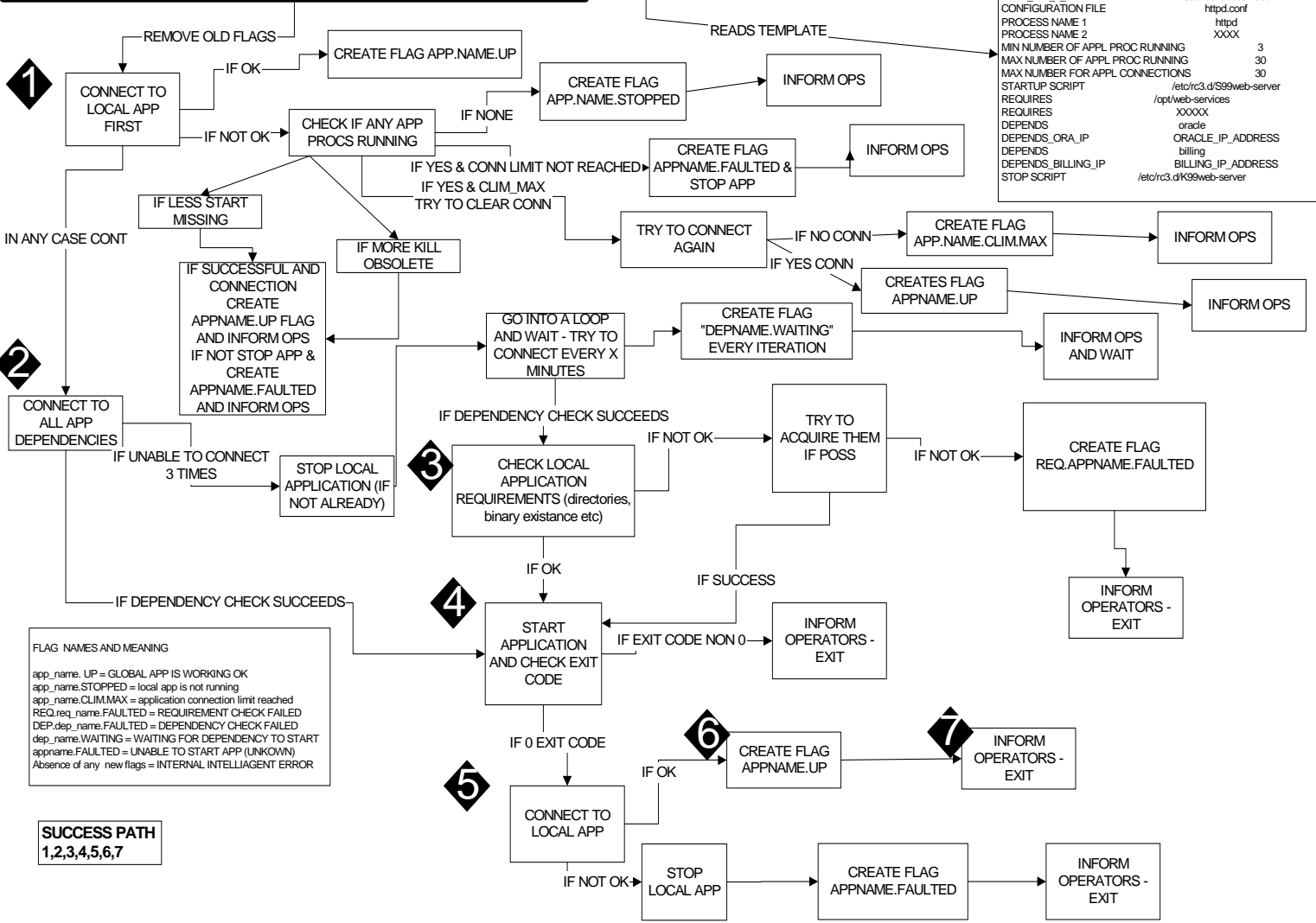
Vladimir Getov

V.S.GETOV@westminster.ac.uk

LOCAL APPLICATION INTELLIAGENT MONITORS LOCAL APP COMPONENT EVERY 15 MINS

```

TEMPLATE
SERVER NAME                zeus
SERVER IP ADDRESS          XXX.XXX.XX.XX
APPLICATION RUNNING        httpd
CONFIGURATION DIRECTORY   /opt/web-services
DATA_DIR_1_NAME            /apps/dir1
DATA_DIR_1_MINT            Rea:/vol/volroot/webdir
CONFIGURATION FILE        httpd.conf
PROCESS NAME 1             httpd
PROCESS NAME 2            XXXX
MIN NUMBER OF APPL PROC RUNNING 3
MAX NUMBER OF APPL PROC RUNNING 30
MAX NUMBER FOR APPL CONNECTIONS 30
STARTUP SCRIPT             /etc/rc3.d/S99web-server
REQUIRES                   /opt/web-services
                           XXXXX
DEPENDS                    oracle
DEPENDS_ORA_IP             ORACLE_IP_ADDRESS
DEPENDS_BILLING_IP        billing
DEPENDS_BILLING_IP        BILLING_IP_ADDRESS
STOP SCRIPT                /etc/rc3.d/K99web-server
    
```



FLAG NAMES AND MEANING

app_name.UP = GLOBAL APP IS WORKING OK
 app_name.STOPPED = local app is not running
 app_name.CLIM.MAX = application connection limit reached
 REQ.req_name.FAULTED = REQUIREMENT CHECK FAILED
 DEP.dep_name.FAULTED = DEPENDENCY CHECK FAILED
 dep_name.WAITING = WAITING FOR DEPENDENCY TO START
 appname.FAULTED = UNABLE TO START APP (UNKNOWN)
 Absence of any new flags = INTERNAL INTELLIAGENT ERROR

SUCCESS PATH
1,2,3,4,5,6,7

