Biblioteca Ştiinţifică Medicală

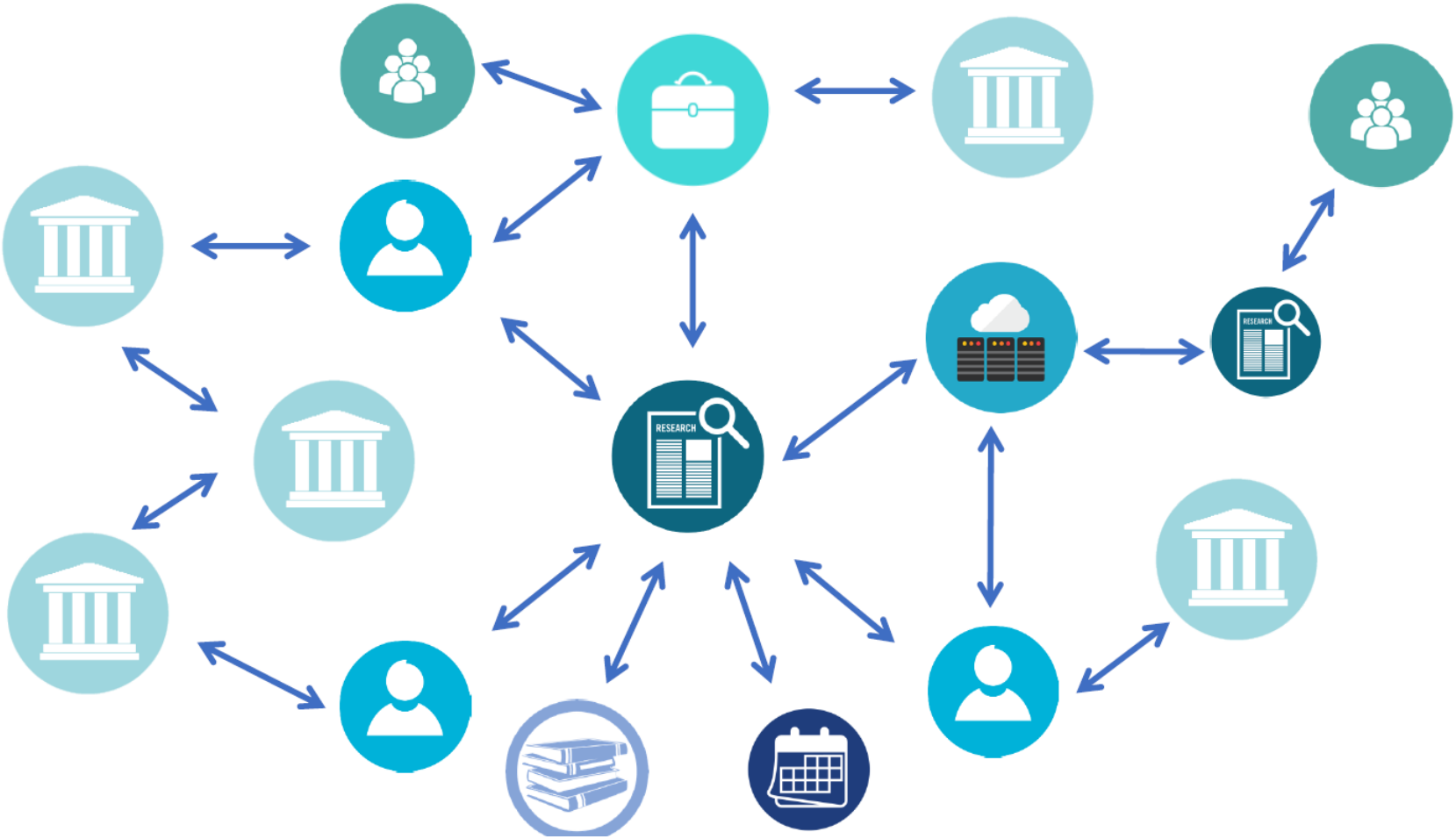
ON ADAPTATION OF THE DSPACE-CRIS INFORMATION SYSTEM

Alexandru Colesnicov, Ludmla Malahov
acolesnicov@gmx.md, lmalahov@gmail.md

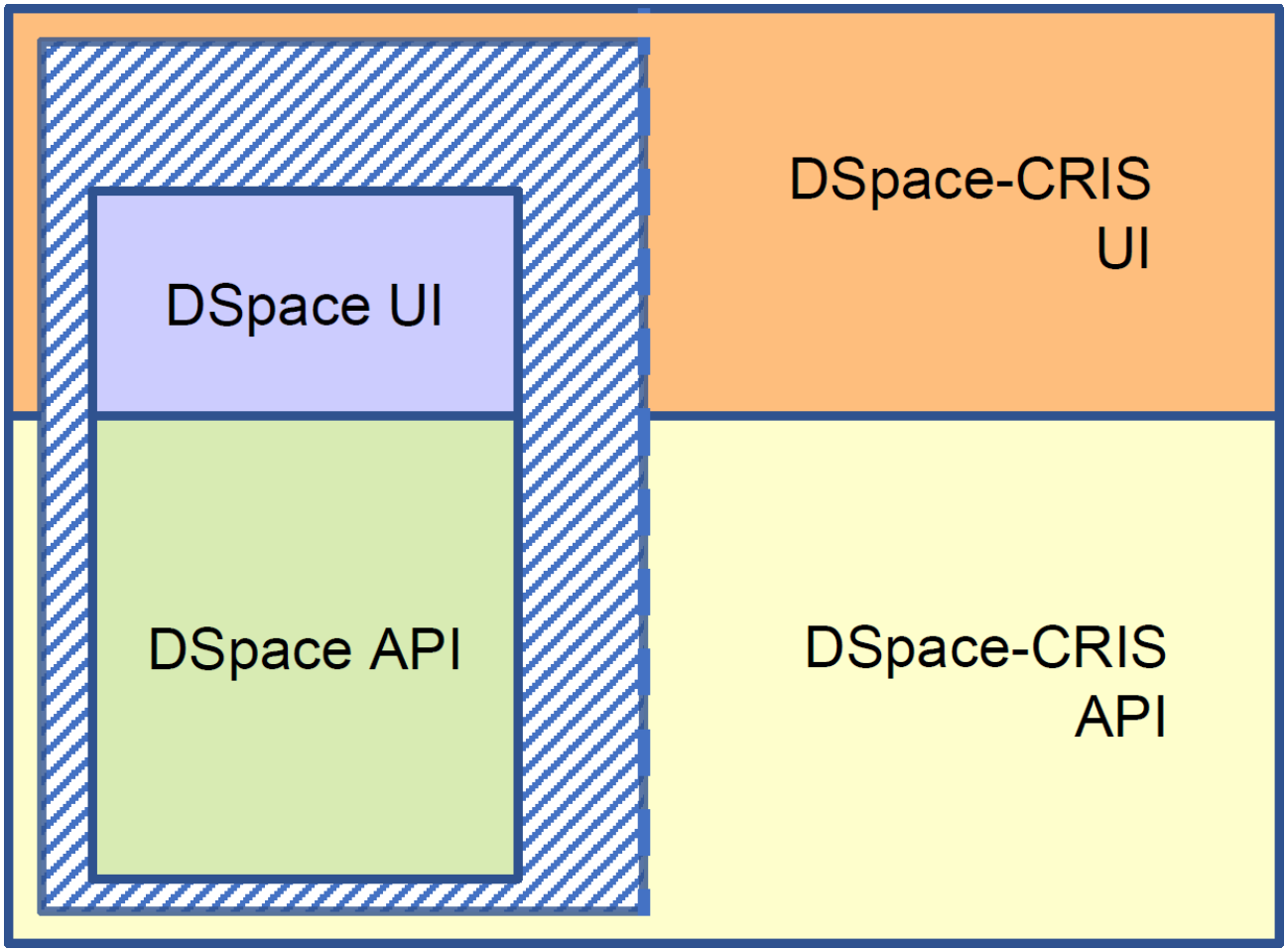
DSpace-CRIS

- A free open source research management system
- Data model represents CERIF objects and relationships: Researchers, Bodies, Projects, Results (publications, data sets, etc.)
- This report discusses problems arising at adaptation of DSpace-CRIS

Research ecosystem



CRIS over DSpace



New approaches to adaptation

- Convergent infrastructure: the whole platform is a conglomerate of independent (sub)systems communicating through net protocols and APIs
- Unification and standardization: a condition sine qua non
- Openness of systems: all protocols and APIs should be available and exhaustive
- Stability and reliability: one of main purposes of platform development

DSpace-CRIS: a convergent system

- Web-interface through Tomcat application server
- JDK
- DB (Postgresql or Oracle)
- SMTP (occasional outgoing mails)
- Any correction and reconfiguration needs Maven+Ant to recreate the system
- Statistics
- OAI and REST acess
- OCRID access
- Support: daily and other indexing, backup/restore, security keys, filename normalization

Virtualization

- Convergent systems are extremely complicated for installation, (re)configuration and support
- A good practice is to install each component in a separate clean environment (sandbox) with fixed set of interactions
- These separate environments may be virtual

Docker

- Full-scale virtual machines are rather cumbersome
- Docker provides lightweight virtual environments (Docker containers) for system components
- Docker-compose is the extension that integrates several docker containers
- In our case, at least three containers will run: these with Tomcat, DB and mailer

Docker service, images and containers

- Docker
  - a server that executes all operations
  - two interfaces (CLI and visual)
  - Docker-compose container integrator
- Ready-made system components are stored as Docker images; a big repository of those Docker Hub is available for free
- Docker container is a running entity of a Docker image

Docker: containers at work (1)

- Docker isolates containers from host OS (Linux, MacOS, Windows 10+)
- **Image:** immutable collection of *layers* that represent differences in the states of the filesystem
- **Container:** the running **image**; an additional *writable layer* reflects changes (results of run)
- Disk memory: is modeled on host disks, on another machine, or in the cloud
- Container can access host networks either through port forwarding or directly

Docker: containers at work (2)

- Container may be stopped and restarted; if image recreates the container, all changes are lost
- To keep changes:
  - Store them in the host filesystem using direct access to host disks
  - Backup them by dedicated utilities
  - Froze the container state with the `*docker commit*` command that creates a new image with one more layer: in the new image, all changes (the formerly writable layer) form a new immutable layer
- Version tracking: similar to GitHub, with the ability to rollback
- Docker-compose: launches a group of containers; a default internal network connects these containers; more virtual networks can be defined

Conclusions

- Docker make installation and support of a complex software platform much easier
- To install the platform anew, the sysadmin installs Docker, copies data stored on the host, and restarts all containers
- All adaptations and reconfigurations are restricted with their specific images
- Adaptations are fixed with one command (`*docker commit*`)

Thank you for your attention!