DISTRIBUTING SOFTWARE APPLICATIONS
BASED ON RUNTIME ENVIRONMENT

N. Ratnikova, D.Evans, Fermilab
R. Darwish, Fermilab
T. Wildish, Princeton
N.Darmenov, CERN

Runtime Environment

Reference Host:

RTE = OS RTE + Appl RTE

- executables ($PATH)
- *.so ($LD_LIBRARY_PATH)
- data files
- flags

compatible / *DAR*

Destination Host:

RTE = OS RTE + Appl RTE

- flags
- data files
- *.so ($LD_LIBRARY_PATH)
- executables ($PATH)

DAR brings necessary files and reproduces the application's runtime environment on the destination host.

Fine Tuning

- allows to exclude files and directories matching specified patterns, e.g.: lib*.a;*.pyc;*.tgz;*.PDF
- allows to preserve files matching a specified pattern
- allows to ignore contents for specified variable names
- interoperates with SCRAM (CMS build tool)
- flexibility in DARball name and architecture identifiers

Additional DAR Features

- extensive validation features
- allowes quick test to verify DARball before distribution
- stores information about DARball creation as metadata
- shows DARball metadata without installation
- provides Python API's for interoperability with other tools
- platform independent

Concept

Distribution of experiment-specific software is a challenging task due to large number of products, versions and cross dependencies. In order to run a single application, it is often sufficient to create a proper runtime environment, and to ensure the availability of required shared objects and data files.

The idea of distributing software applications based on runtime environment is employed by DAR (Distribution After Release) tool. DAR allows to automatically replicate application's runtime environment based on the reference software installation. Assuming that software is relocatable, applications are packaged into a self-contained DARball, which can be installed and executed on any computing node with a compatible operating system.
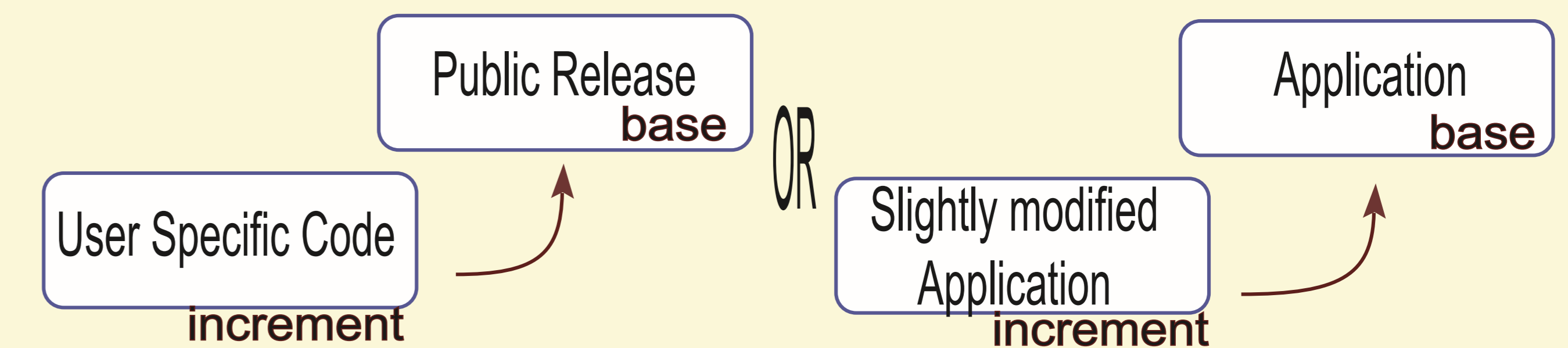
References

CME Experiment: http://cmsdoc.cern.ch/cms/outreach/html
N. Ratnikova, A. Afaq, G. Graham, T. Wildish, V. Lefebure:
Software Packaging with DAR. Nucl.Instrum.Meth.A534:110-114,2004
N. Ratnikova, A. Sciaba, S. Wynhoff:
Distributing Applications in Distributed Environment.
Nucl.Instrum.Meth.A502:458-460,2003
N. Ratnikova, G. Graham:CMS Software Distribution and Installation
Systems: Concepts, Practical Solutions and Experience at Fermilab
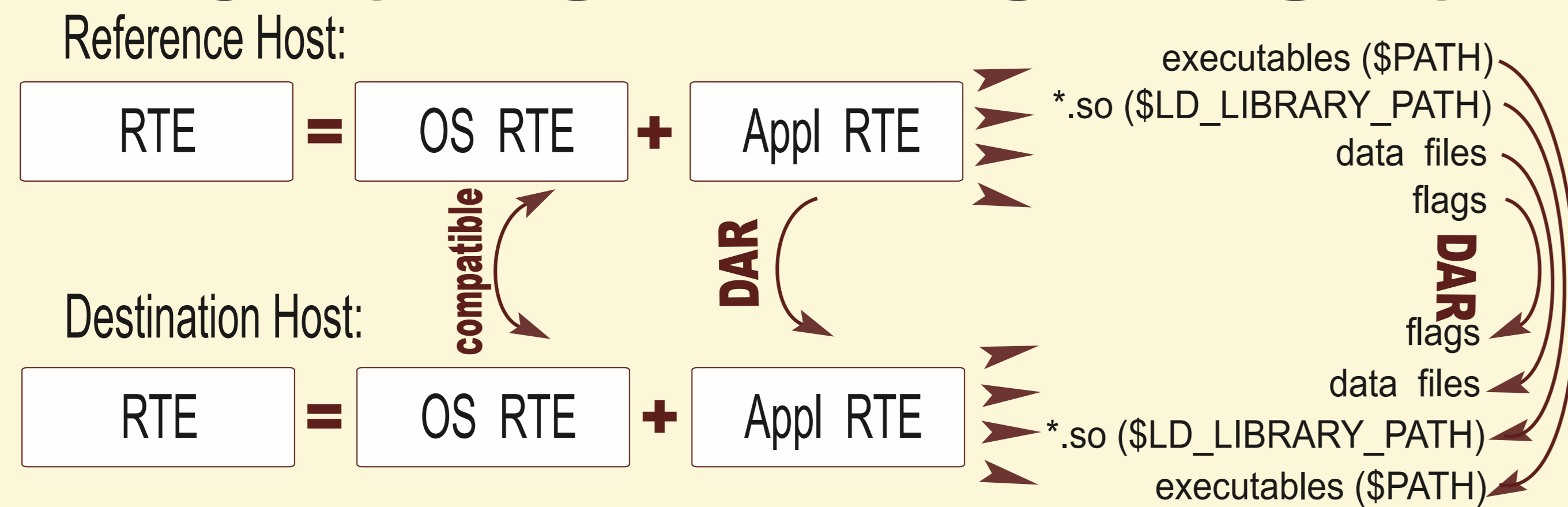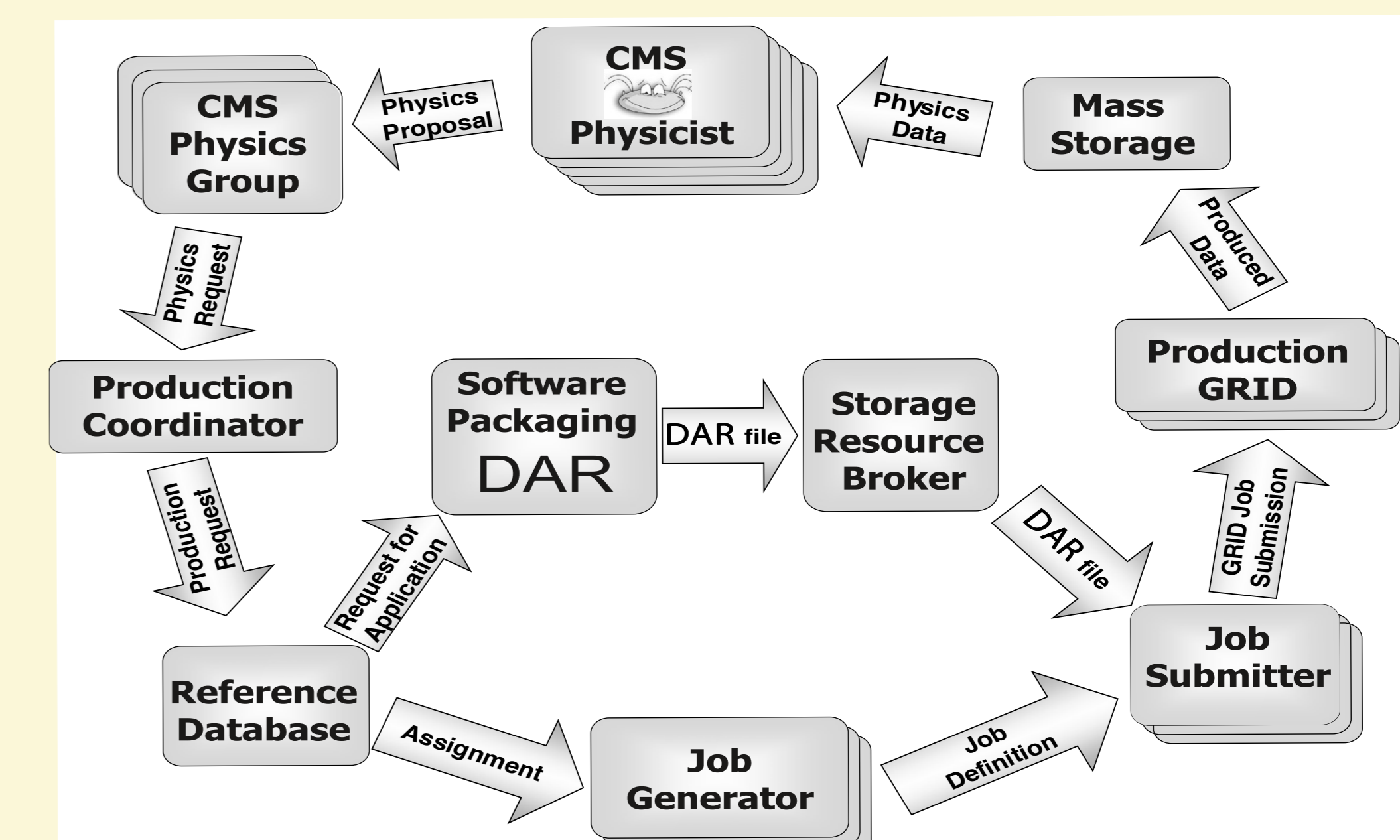as a CMS Tier 1 Center.CHEP01, Beijing, September, 2001

For more information contact:
Natalia M. Ratnikova,
Fermilab, Computing Division, CMS department
Email: natasha@fnal.gov

Incremental Distributions

- contain complete information about the runtime environment.
Files that are common with the base DARball (identical relative path, name and md5sum) are replaced by a reference to the file in the base DARball. This allows to re-use existing installations:

Public Release **base**

User Specific Code *increment*

OR

Application **base**

Slightly modified Application *increment*

Uses

DARballs are light-weight distributions and can be used anywhere where software does not need to be rebuilt. They are most suitable for use on opportunistic GRID resources, as they allow to avoid excessive efforts of complete installation of experiment-specific software.
For over three years, DAR is used in CMS for worldwide Monte Carlo production:

Monte Carlo Production Cycle

DAR²