BEgrid Seminar, 16th October 2007

Integrating P2P file sharing with P2P Grid computing

Cyril Briquet
Xavier Dalem

Department of EE & CS
University of Liège, Belgium

Overview

- What is a P2P Grid ?
- BitTorrent
- Data Management
- Scheduling Tasks with large files
- Exploiting Data Redundancy
- Deployment
- Conclusion

Overview

- **What is a P2P Grid ?**
- BitTorrent
- Data Management
- Scheduling Tasks with large files
- Exploiting Data Redundancy
- Deployment
- Conclusion

What is a P2P Grid ?

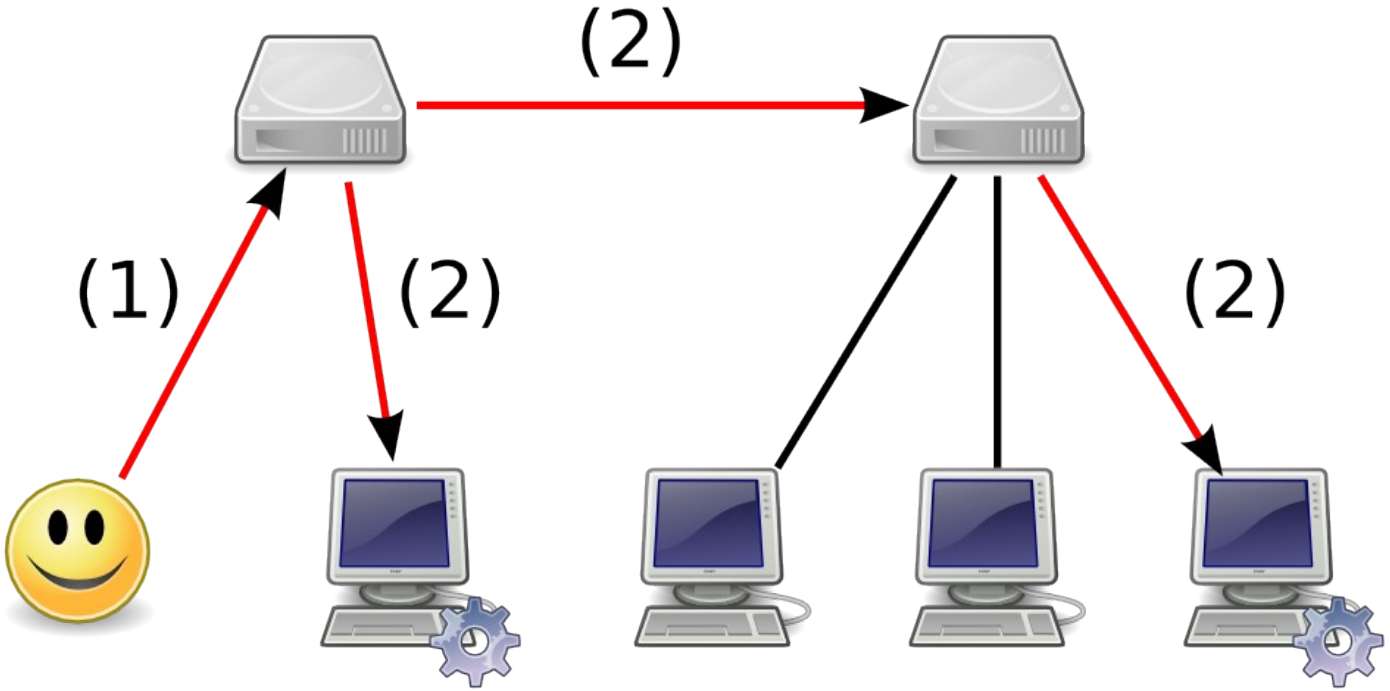
- Site administrators group their Resources to aggregate computational power for site Users  
- A Peer  manages Resources of a site

  Users submit sets of independent computational Tasks (**Bags of Tasks** or **BoT)** to the Peer at their site 
- Application domains : GIS, computer vision, data mining, bioinformatics

What is a P2P Grid ?

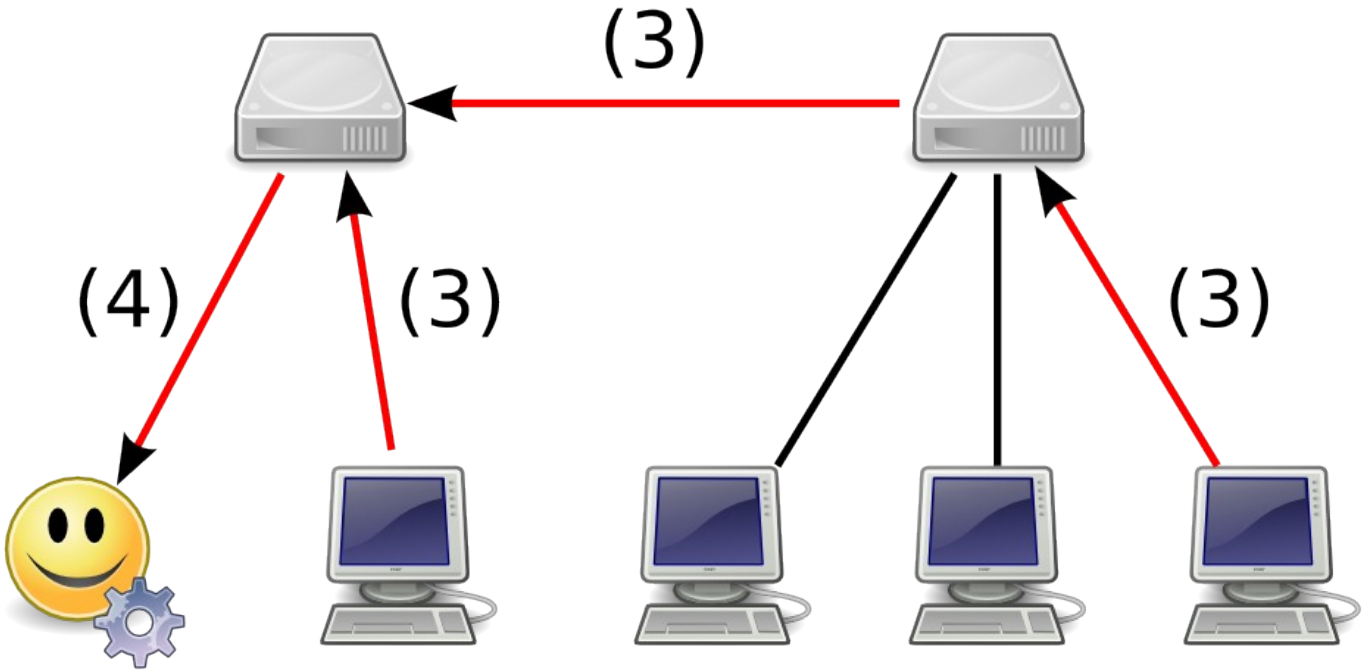
- **User** 
- **Peer** 
  - controls a set of Resources 
  - has a storage service (data cache) 
- **Resource** 
  - edge computer
  - runs Tasks (J2SE 5.0 code) 
  - has a storage service (data cache) 

What is a P2P Grid ? - Task submission



- Peers **submit** Tasks **to their Resources**, or **to other Peers**

What is a P2P Grid ? - Task results



- Results are uploaded back to the User

What is a P2P Grid ? - Resource exchange

- Peers act in their own interest, but may **cooperate by exchanging computing time**

- **Bartering** = fully distributed, moneyless exchange and accounting (as opposed to the Grid economy)

What is a P2P Grid ? - File transfers

- Tasks may require large input data files, better not transfer these alongside

- Centralized protocols (e.g. FTP) not well adapted to P2P Grids

- Idea: combining P2P Grid with a P2P transfer protocol

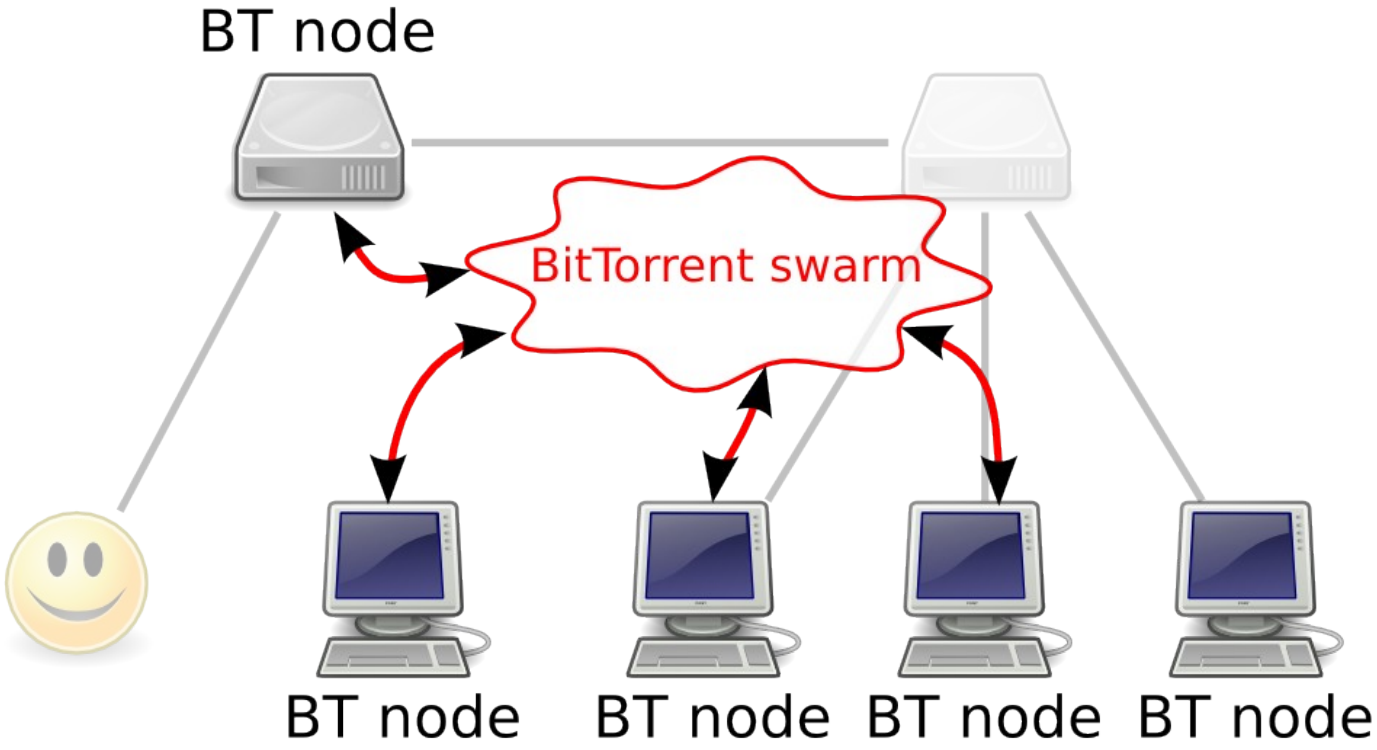
Overview

- What is a P2P Grid ?
- **BitTorrent**
- Data Management
- Scheduling Tasks with large files
- Exploiting Data Redundancy
- Deployment
- Conclusion

BitTorrent

- BitTorrent = P2P file sharing protocol
- Aims:
  - Reliability
  - Speed
  - Scalability : reduced load on sharer (seeder)

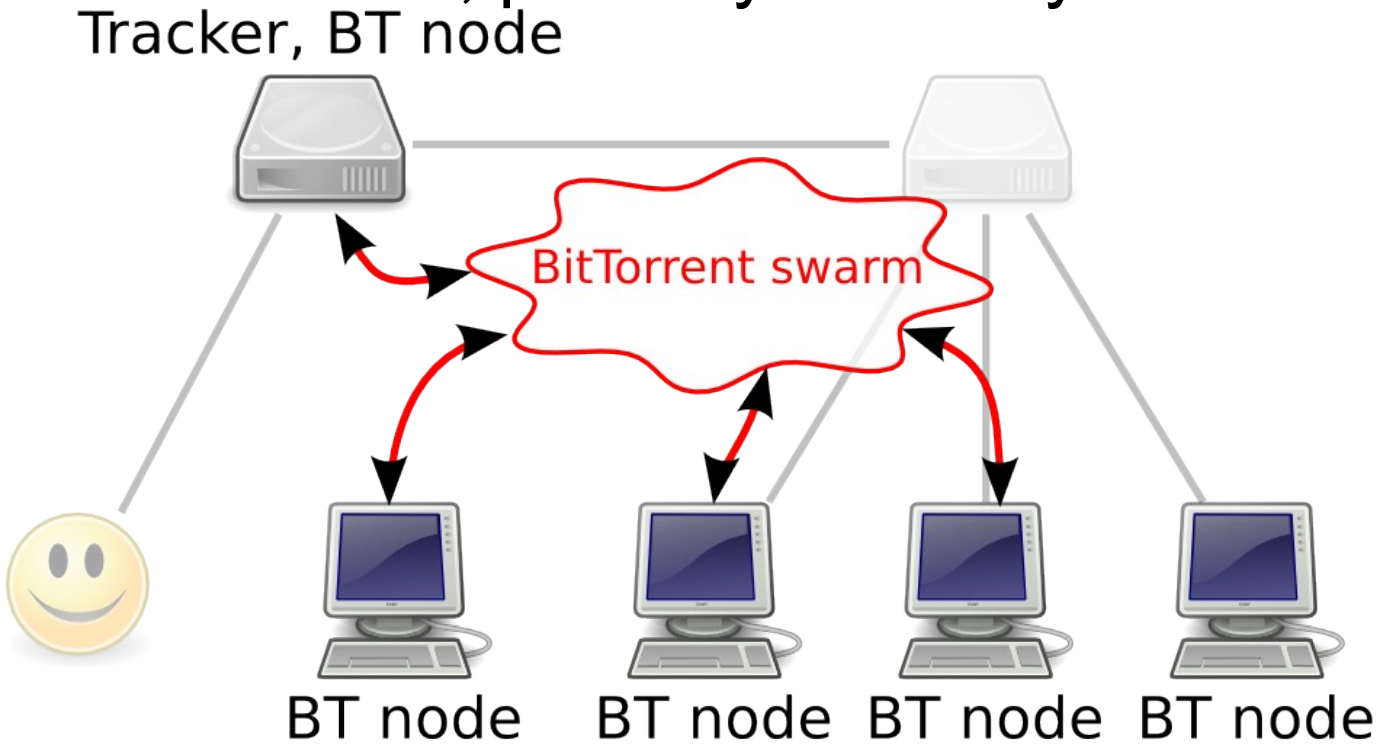
BitTorrent - Key Definitions



- BitTorrent client on Grid Peers **and** Resources (we'll call them **BitTorrent Nodes**)
  → they **exchange data pieces** with one another

BitTorrent - Key Definitions

- 1 **BitTorrent tracker** coordinates transfers between BT Nodes, possibly for many files



- each Grid Peer runs a BitTorrent tracker to support the distribution of his own files

BitTorrent - Interesting Properties

- BitTorrent is able to exploit network **links between BT Nodes** (so-called **orthogonal bandwidth**)

  → cost of multiple simultaneous transfers of a file
  ~ cost of 1 transfer of that file

- BitTorrent is able to adapt to very **dynamic network conditions**

  → important in P2P Grids

BitTorrent - Interesting Properties

- downloads come from multiple Nodes at once
  → BitTorrent is most efficient with lots of Nodes
  (so-called *flash crowds*)

- when a BT Node has finished to download a file,
  it continues to share the file
  (default behavior,
  keeps a high number of sharing sources)

BitTorrent - How to share a file ?

Steps for a Consumer Peer to share a file:

- start a tracker (or use the running one)
- create a torrent metadata file
  (contains tracker URL, ...)
- start a BitTorrent client to share the file
  (= become a BT Node for that file)
- publish the torrent metadata file

BitTorrent - How to download a file ?

Steps for a Resource to download a file:

- obtain the torrent metadata file
- connect to the tracker
- initially download a few pieces
  from some BT Node sharing the complete file
- invite other BT Nodes to provide pieces
  by contributing pieces
- give preference to Nodes with good bandwidth
  and rare pieces

Overview

- What is a P2P Grid ?
- BitTorrent
- **Data Management**
- Scheduling Tasks with large files
- Exploiting Data Redundancy
- Deployment
- Conclusion

Data Management - Transfer Trigger

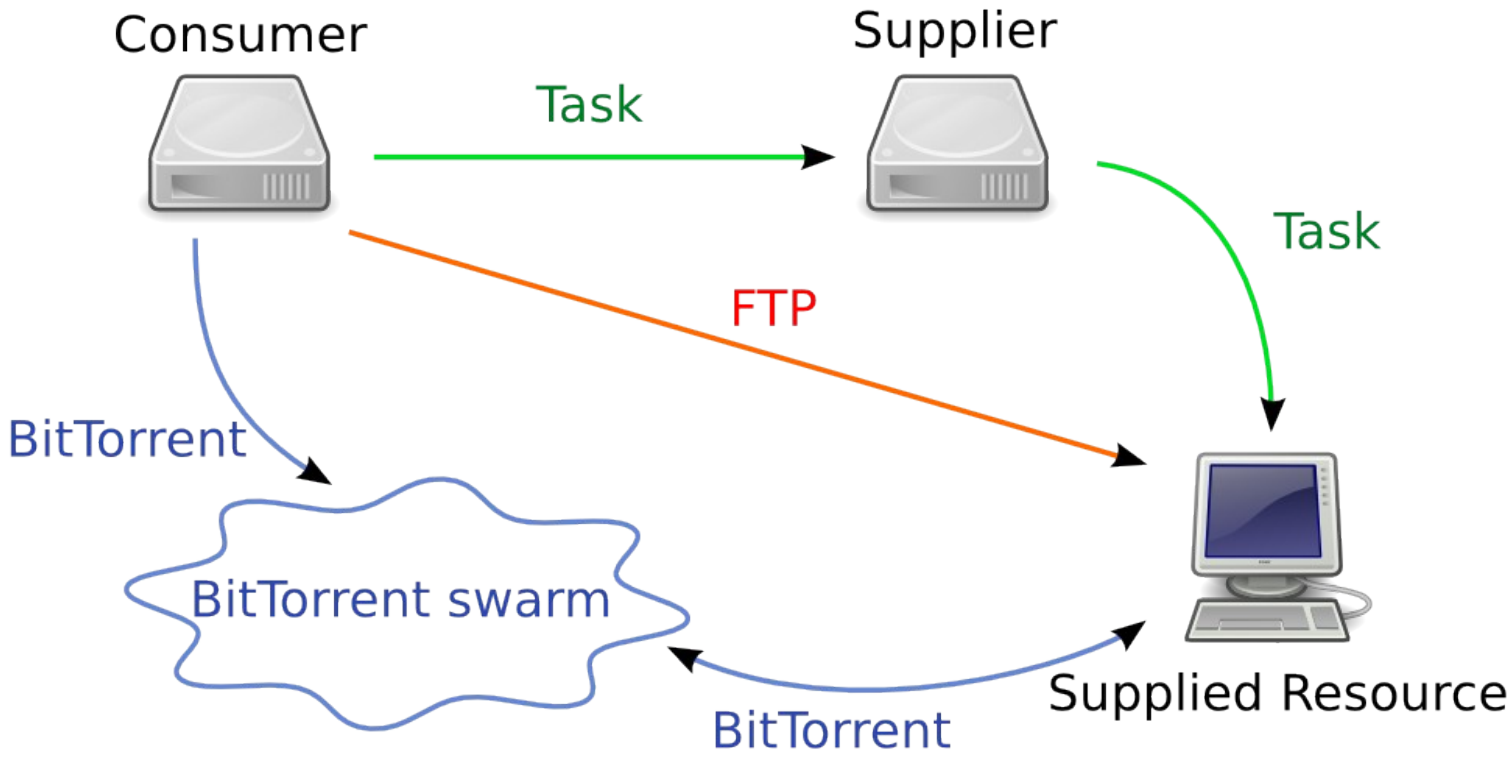
Even with an efficient downloading protocol, downloading data increases Task response time :

- each Resource keeps a **cache** of recently downloaded and used data files

- when a Task is scheduled to a Resource, the Resource **only downloads uncached** data **files**

Data Management - Typical Transfer

there are multiple data paths between consumer peer and supplied resource:



Overview

- What is a P2P Grid ?
- BitTorrent
- Data Management
- **Scheduling Tasks with large files**
- Exploiting Data Redundancy
- Deployment
- Conclusion

Scheduling Tasks with large files

- Data-Intensive BoT (D-I BoT) =
  BoT where Tasks process big data files

- data files of a D-I BoT may be redundant,
  i.e. some/all Tasks process the same files
  (e.g. parameter sweeps)

Scheduling Tasks with large files

Things-to-do when scheduling D-I BoT:

- schedule Tasks where **data** is **present**
- **cache** data on Resources
- **proactively replicate** data to data caches

Scheduling Tasks with large files

- **Spatial Tasks Grouping** =
  schedule data-sharing Tasks sequentially
  to the same Resources
  → **data locality ++, less transfers**
  → **execution parallelism --**

- **Temporal Tasks Grouping** =
  schedule (data-sharing) Tasks concurrently
  to multiple Resources
  → **temporal locality ++, more transfers**
  → **execution parallelism ++**

Overview

- What is a P2P Grid ?
- BitTorrent
- Data Management
- Scheduling Tasks with large files
- **Exploiting Data Redundancy**
- Deployment
- Conclusion

Exploiting Data Redundancy

Many files of a BoT may be identical

- transfer them with BitTorrent
- but... BitTorrent is most efficient to handle flash crowds

Exploiting Data Redundancy

Idea: have many Grid nodes
download the same file concurrently

- maximize simultaneous scheduling of Tasks depending on identical data files
- flash crowds created *on demand*,
in a controlled way

Exploiting Data Redundancy

- sometimes, Tasks depending on identical data files cannot be scheduled concurrently
  (e.g. there are not enough resources simultaneously available)
- some data files may also be required by multiple BoT spread over time

Exploiting Data Redundancy

- data caching by Resources

and

- Peer/Resource data-aware scheduling

enable to reuse data files and
avoid unnecessary data transfers

Overview

- What is a P2P Grid ?
- BitTorrent
- Data Management
- Scheduling Tasks with large files
- Exploiting Data Redundancy
- **Deployment**
- Conclusion

Deployment

- Lightweight Bartering Grid (LBG) middleware
- implemented in Java (J2SE 5.0)

- discrete-event simulator also available
- released as Free and Open Source Software (GPL license)
- uses only Free and Open Source libraries (Apache FTP server, Azureus, edtFTPj)

Deployment - Required Software

- each Peer: BT tracker, BT client, FTP server
- each Resource: BT client, FTP client



Data Manager	Peer (Consumer)	Peer (Supplier)	Resource
data storage			
BT data tracking			
BT data sharing			
FTP data sharing			
BT data downloading			
FTP data downloading			

Overview

- What is a P2P Grid ?
- BitTorrent
- Data Management
- Scheduling Tasks with large files
- Exploiting Data Redundancy
- Deployment
- **Conclusion**

Conclusion

- cooperation between P2P technologies (for both computing and data transfers)
- highly scalable data transfers architecture:
  - network load reduced (caching)
  - better spread (BitTorrent)
- easily deployable implementation

Conclusion – Technical Remarks

- BitTorrent activated whenever possible (for small files → FTP)
- flash crowds created *on demand*
- caching always activated (configurable size to limit storage cost)
- data-aware scheduling
- execution & data transfers parallelism both enabled

BEgrid Seminar, 16th October 2007

Thank You !

Cyril Briquet
Xavier Dalem

Department of EE & CS
University of Liège, Belgium

Article reference

C. Briquet, X. Dalem, S. Jodogne and P.A. de Marneffe.

Scheduling Data-Intensive Bags of Tasks in P2P Grids with BitTorrent-enabled Data Distribution.

In *Proc. UPGRADE-CN'07, HPDC Workshops*, Monterey Bay, CA, USA, 2007.

Data Management - Protocol Selection

Each data file is statically tagged
with a **protocol preference** at BoT submission time

	Weakly shared (< 10 Tasks / BoT)	**Strongly shared** (≥ 10 Tasks / BoT)
File size > 50 MB	**BitTorrent**	**BitTorrent**
20 MB ≤ File size ≤ 50 MB	**FTP**	**BitTorrent**
File size < 20 MB	**FTP**	**FTP**

BitTorrent support in Grids

- clusters: 1X (loose coupling + no paper)
- Desktop Grids: 2X
  (deep coupling + modified BT) (loose coupling)
- Volunteer Grids: under preparation
- *regular* Grids: 2X (preliminary work, simulations)
- P2P Grids: this paper ! (deep coupling)