

Nomes em sistemas distribuídos



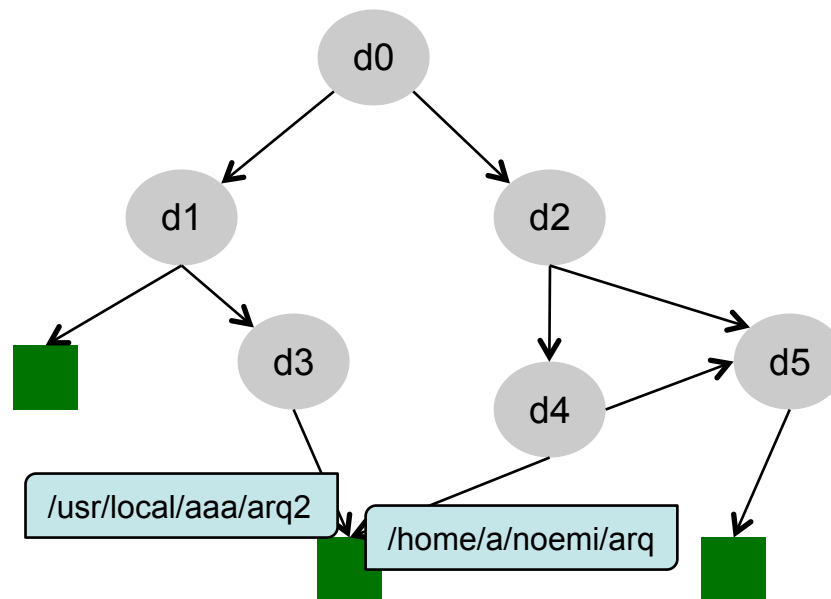
conceitos básicos

- nome: cadeia de bits ou caracteres
- nomes, nomes puros, identificadores
- nomes legíveis por sistemas x por usuários
- nomes estruturados e não estruturados



espaços de nomes

- grafos acíclicos com entidades nos nós “folha”



resolução de nomes

- mapeamento nomes → entidades
 - mapeamento para *atributos*
 - endereço
 - propriedades
- resolução iterativa
 - resolvedor contacta cada nó no caminho
- resolução recursiva
 - primeiro nó contactado passa parte do nome que não lhe diz respeito em consulta a outro nó
 - uso de caches



resolução de nomes

- resolvidor pode ser estruturado de várias formas:
 - processo cliente
 - kernel cliente
 - processo independente
 - serviço em separado
- mecanismo de fecho (closure)
 - início da busca



serviço de nomes

- local
 - servidor único
 - local ou remoto
 - exemplo: CORBA
- distribuído
 - múltiplos servidores
 - espaço de nomes é replicado e/ou particionado
 - exemplos: DNS, LDAP

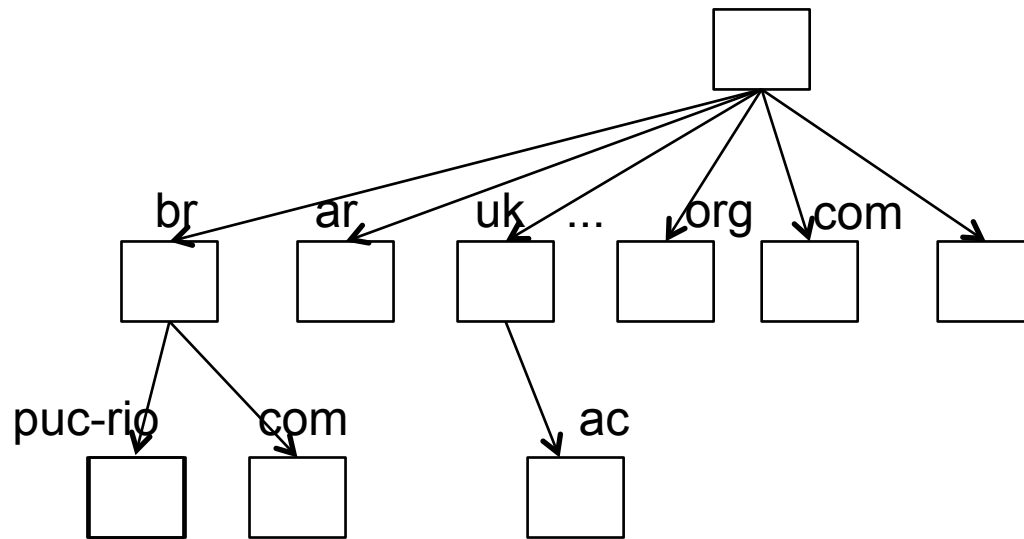


serviços CORBA

- nomes
 - mapeamento para IOR
- trading
 - mapeamento para propriedades



DNS



- consultas iterativas
- replicação e cache
- divisão do espaço em zonas

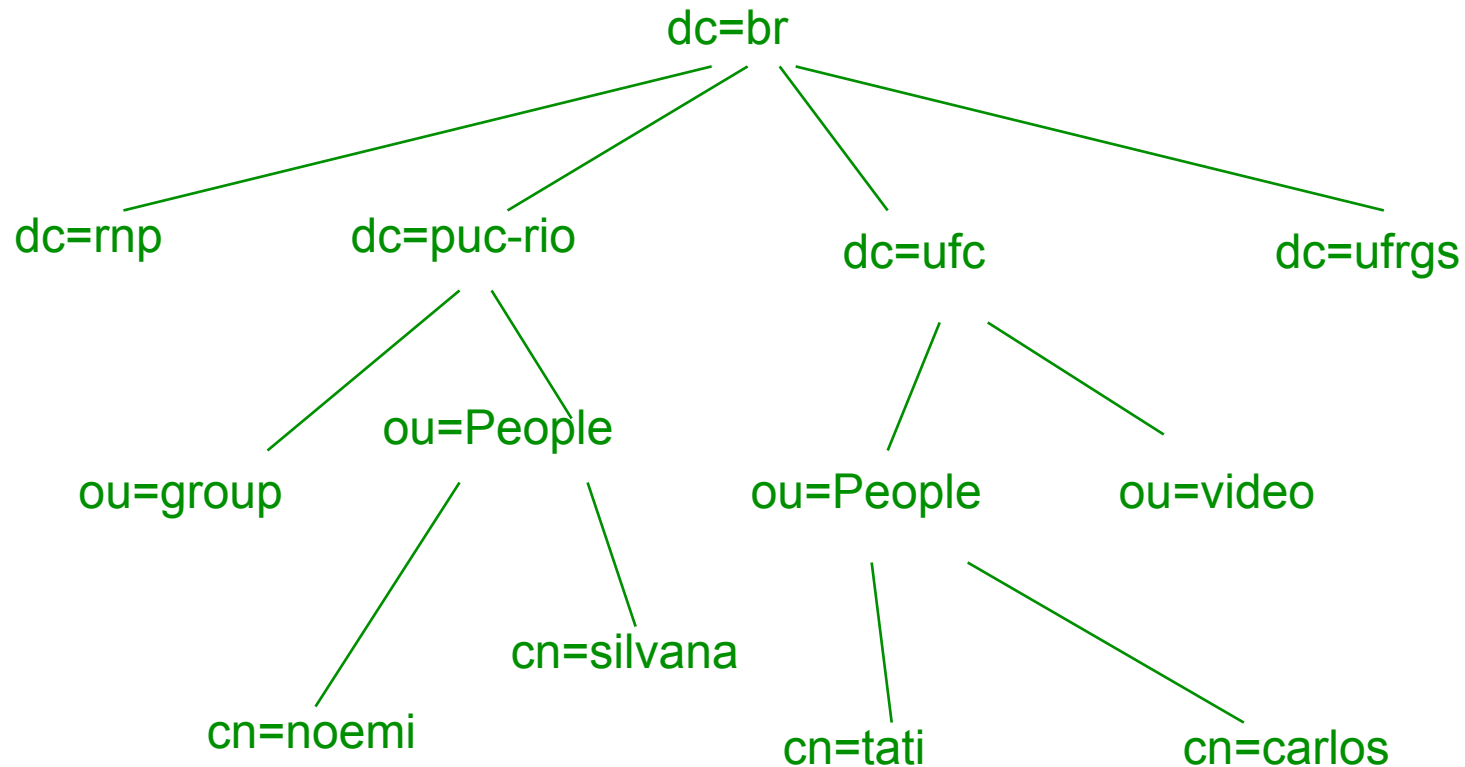


serviços de diretórios

- busca de entidades por atributos
 - nome pode ser um dos atributos
- originalmente chamados de *white pages*
 - origem no X500
- LDAP e openLDAP
- protocolo de acesso por aplicações
 - em particular, para obter certificados



LDAP - modelo de nomes



- `dn: cn=noemi,ou=people,dc=puc-rio,dc=br`



exemplo de entrada

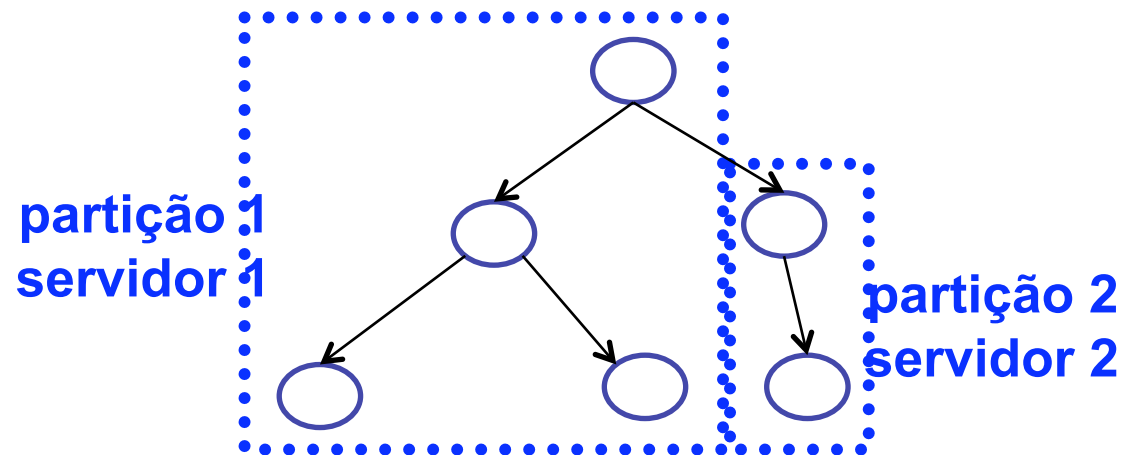
dn: cn=Barbara J Jensen, dc=example, dc=com
cn: Barbara J Jensen
cn: Babs Jensen
objectClass: person
sn: Jensen

- uma entrada pode pertencer a várias classes de objetos



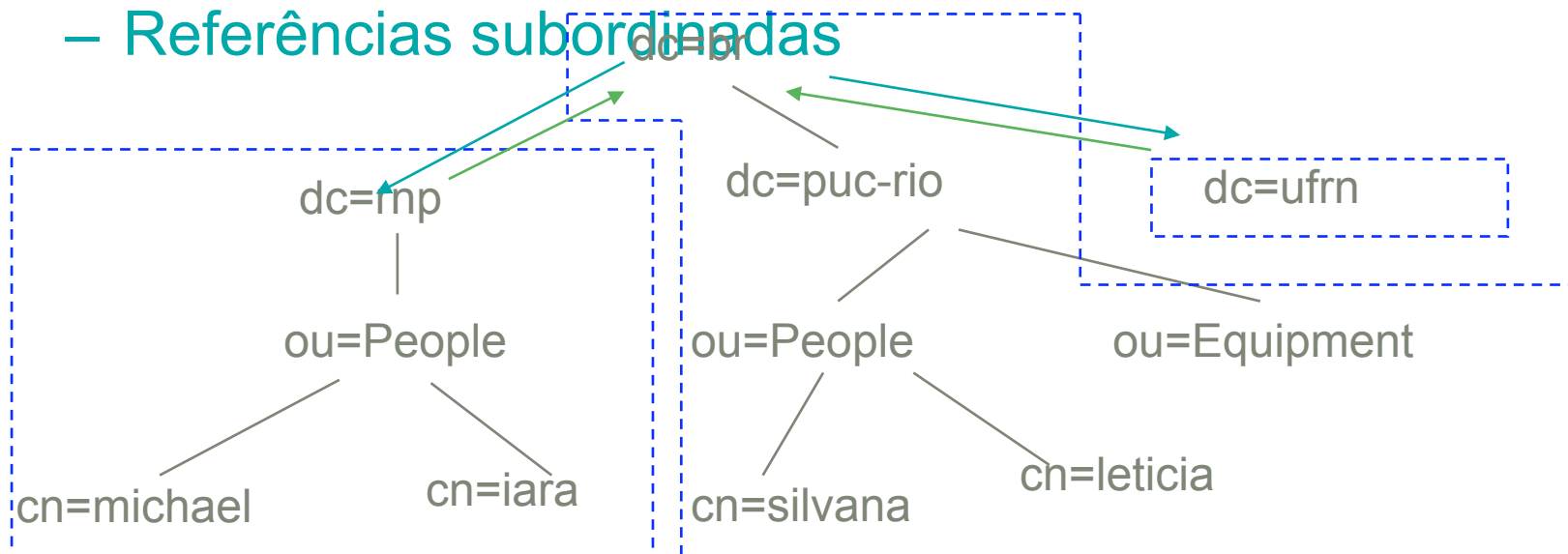
suporte para distribuição

- distribuição de informação (referrals)



unindo as partições

- Referências de conhecimento imediatamente superior
- Referências subordinadas



métodos para unir as partições

- Referrals
 - resolução iterativa
 - funções de APIs fazem acesso aos diferentes servidores automaticamente.
- Chaining
 - resolução recursiva

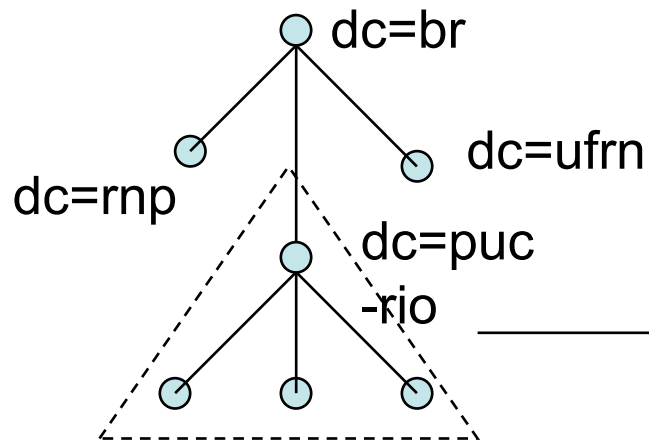


replicação

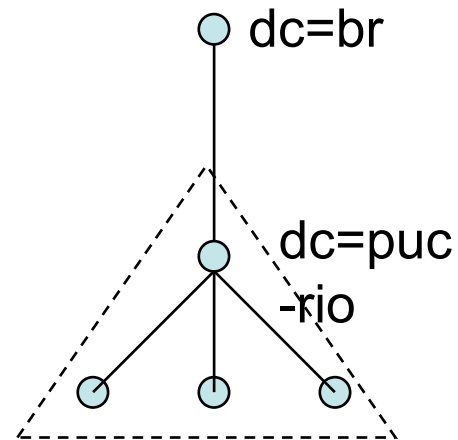
- openldap: cópia do conteúdo do diretório
 - Total ou parcial (sub-árvore)
- objetivos: desempenho e a confiabilidade
 - Aproxima o dado dos usuários
 - Distribui melhor a carga dos servidores
 - Garante redundância do serviço, minimizando os efeitos de falha no servidor ou queda na rede



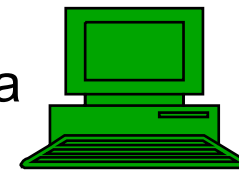
replicação de uma Sub-Árvore



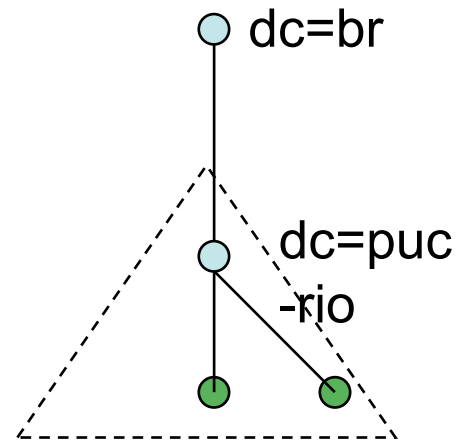
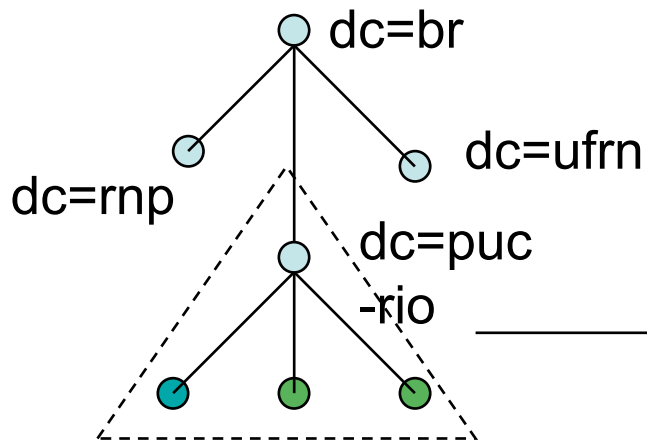
Mestre



Réplica

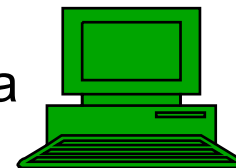


replicação de tipos de entrada



Mestre

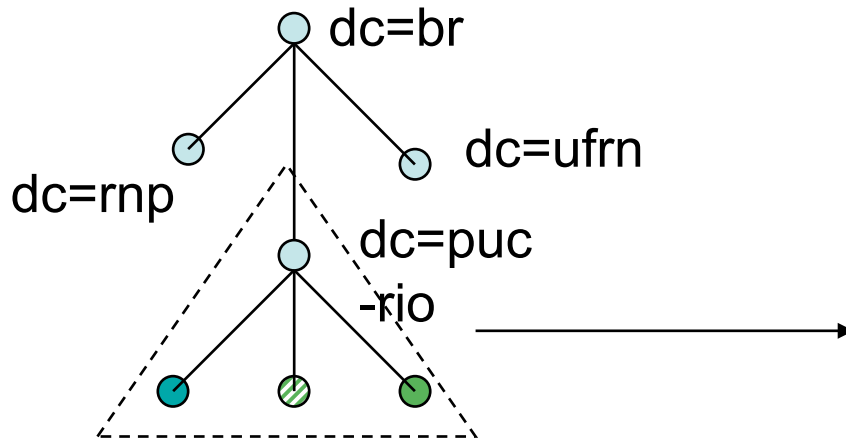
Réplica



- objectclass=organization
- objectclass=printer
- objectclass=person

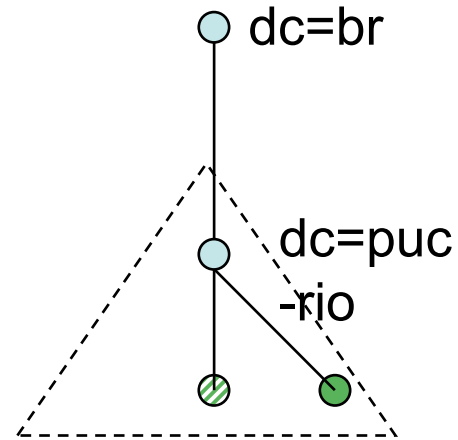


replicação de atributos

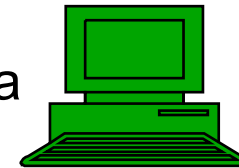


Mestre

- cn=Noemi Rodriguez
homephone: 25409807
mail: noemi@inf.puc-rio.br



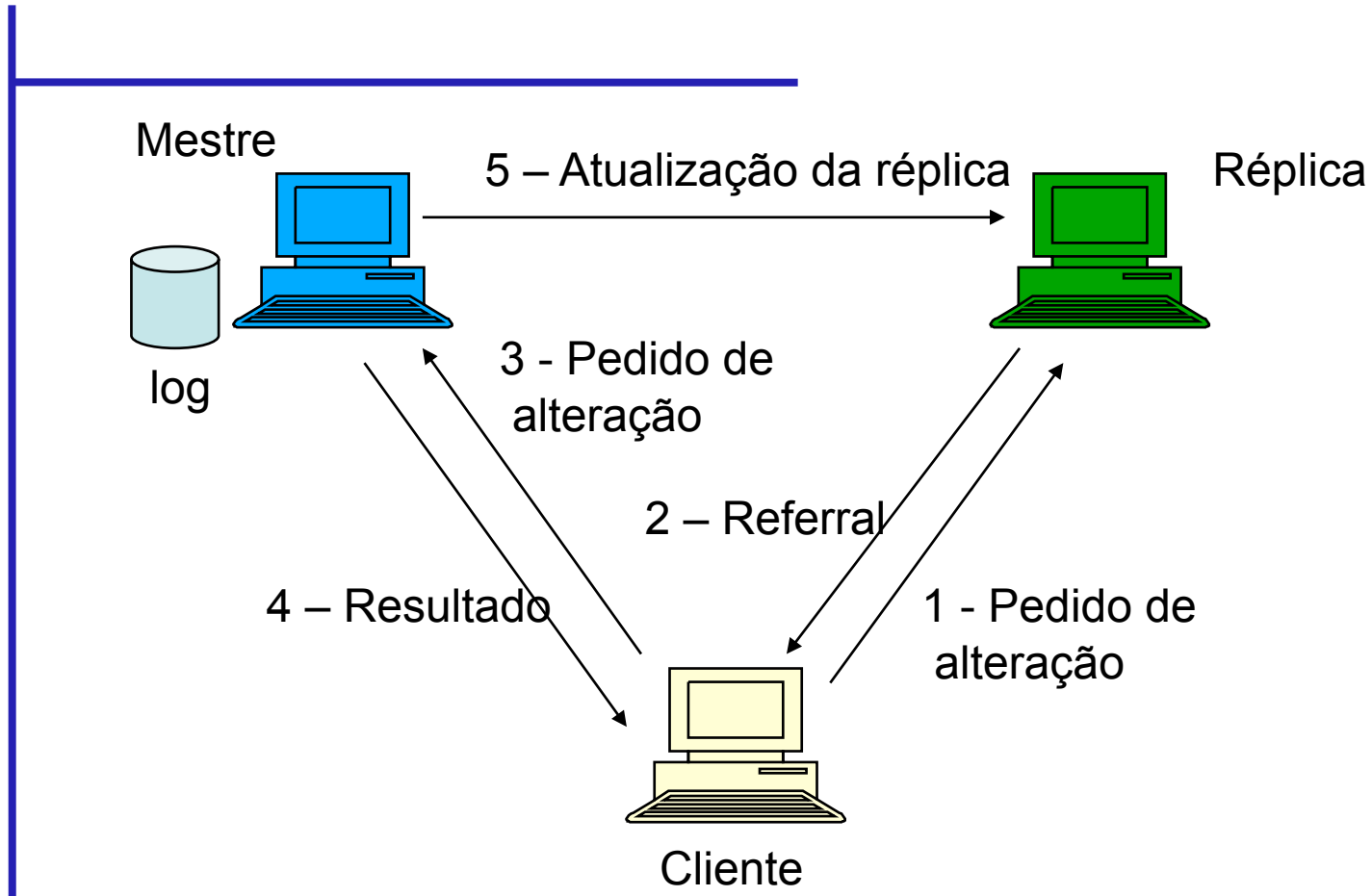
Réplica



- cn=Noemi Rodriguez
mail: noemi@inf.puc-rio.br



escrita em réplicas



esquema brEduPerson

```
dn = cn=esdras, ou=peessoas, dc=ufc, dc=br
objectclass = inetOrgPerson
objectclass = brPerson
objectclass = schacPersonalCharacteristics
Mail = esdras@...
HomePostalAddress = ...
userCertificate = "..."
```

```
dn = brRId=biologia, cn=esdras, ou=peessoas, dc=ufc, dc=br
objectclass = brEduRelatedInfo
objectclass = brEduPersonAffiliation
brRId = biologia
brEduAlliationType = student
entranceDate = 01/03/99
exitDate = 31/12/02
```

```
dn = brRId=foneEmerg, cn=esdras, ou=peessoas, dc=ufc, dc=br
objectclass = brEduRelatedInfo
objectclass = brVoIP
brRId = foneEmerg
alias = ...
hashIP = ...
```

