

Pastry

Bởi:

Khoa CNTT ĐHSP KT Hưng Yên

Tương tự như Chord, Pastry tạo ra một hệ thống hoàn toàn phân tán

Việc định tuyến được dựa trên số lượng gần gũi của các khóa

Thiết kế Pastry

Pastry node và mục dữ liệu được kết hợp chặt chẽ với m bit nhận dạng

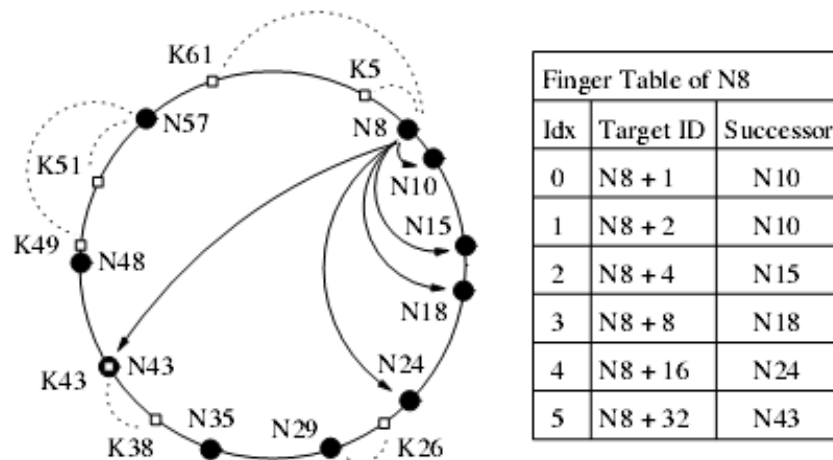
Vùng dữ liệu từ 0 đến $2^m - 1$ (m đặc trưng 128 bit)

Kết hợp 1 khóa được giới hạn 1 node ID hoặc 1 key, tương ứng

Pastry hiển thị mã nhận dạng như một chuỗi số $2b$, b chọn giá trị 4

Khóa A là được định vị tới node ID có số gần nhất

Ví dụ minh họa khoảng mã nhận dạng Pastry với 4 bit nhận dạng và $b=2$



Mô hình mạng Pastry

Pastry

Với $b=2$, vì vậy tất cả các số, các digit nhỏ hơn 4

Node gần nhất được tới có khóa K01 là N01, trong khi K03 được định vị trên node N10

Khoảng cách của khóa K22 tới node N21 và N23 là bằng nhau cho nên chọn khóa thỏa mãn yêu cầu nhất

Các thông số:

$|L|$: là một kiểu cấu hình tham số có giá trị bằng 16 hoặc 32

M: là số nodes thiết lập hàng xóm gần nhất $=2b, 2.2b$

N: là số nodes tham gia vào mạng Pastry

l: là độ dài của nodeID (digit) gán cho mỗi node

b: độ lớn của mỗi digit trong nodeID

Thông tin định tuyến

Routing Table của Pastry được chia làm 3 thành phần chính

+ Rounting table (Bảng tìm đường)

- Lưu trữ những liên kết tới các mã nhận dạng (giống như Finger Table của Chord)

Có $\epsilon \log_2^b(N)$ ù dòng

Mỗi dòng có $2b-1$ điểm vào

Mỗi điểm vào có l digit

Mỗi digit có giá trị lớn không quá $2b$

+ Left set: Những node mà gần nhau nhất trong thời kỳ của mạng cục bộ được liệt kê trong việc thiết lập hàng xóm

Có $|L|/2$ nodeID gần nhất

Được sử dụng trong việc định tuyến các tin nhắn

+ Neighborhood set:

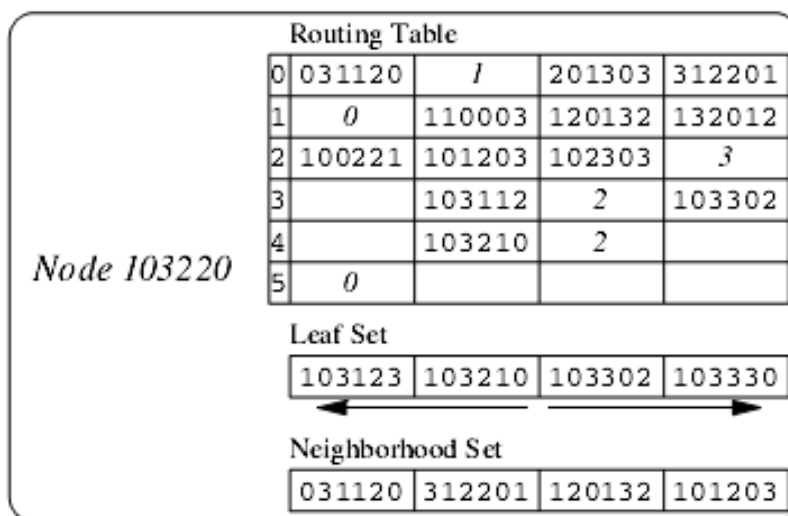
Pastry

Chứa đựng nodeID và địa chỉ IP của $|M|$ nodes gần nhất

Sử dụng trong việc bảo trì các thuộc tính liên quan đến vị trí các nodes

Pastry dựa trên cơ sở mạng xấp xỉ vô hướng hệ mét

Ví dụ: Thông tin về bảng tìm đường với $b=2$



Bảng định tuyến mạng Pastry

Thuật toán tìm đường

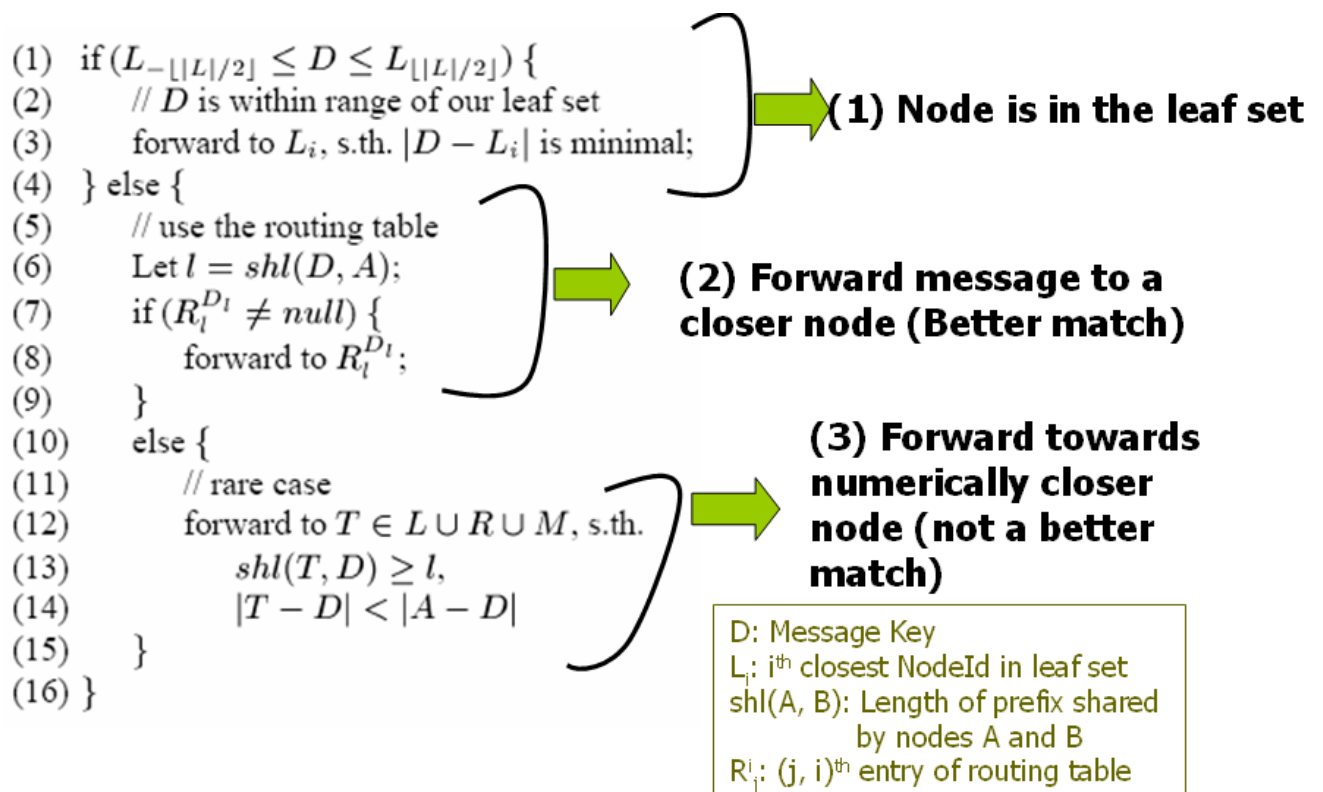
Được sử dụng trong việc định tuyến các tin nhắn có khóa D đến node có ID A

R_l : điểm vào trong bảng tìm đường R ở cột I , hàng l ($0 < i < 2^b$, $0 \leq l \leq \lfloor 128/b \rfloor$)

L_i : nodeID thứ i gần nhất trong mục Left set ($-\lfloor |L|/2 \rfloor \leq i \leq \lfloor |L|/2 \rfloor$)

D_l : là giá trị của digit thứ l trong khóa D

$\text{shl}(A, B)$: chiều dài của tiền tố trong digit giữa A và B



Tự tổ chức và thích nghi mạng

Khi có node mới tham gia mạng:

- Được gán nodeID là X, đó là mã băm SHA-1 địa chỉ IP hoặc public key
- Xây dựng bảng định tuyến, Leftset, Neighbor: Khởi động bảng trạng thái và thông báo với những node khác gần nó, ví dụ node A
- Tự định vị trí nhờ sử dụng "expanding" IP multicast
- Node X hỏi A về bảng định tuyến kết nối truyền X như là một thông báo
- Pastry định tuyến đến node Z gần X nhất
- A, Z trả lời yêu cầu kết nối đến và tất cả các node trên đường tới X gửi bảng trạng thái tới X

X kiểm tra thông tin này, cập nhật bảng định tuyến của nó và thông báo về X.

Khi có node rời khỏi mạng(fail, depart, no warning)

Pastry

- Để sửa chữa node hỏng cần:

- × Thay thế node đã hỏng trong Left set nhờ Neighbor của nó liên lạc với các node tìm ra ID phù hợp nhất update vào Left set (số node hỏng đồng thời $\leq |L|/2$ node gần nhau)
- × Sửa chữa điểm vào Rdl trong bảng tìm đường bằng $R_{i+1} (i \neq d)$ ở một node thích hợp
- × Sử dụng định tuyến tin nhắn liên lạc với mỗi thành viên định kỳ để phát hiện ra node đang tồn tại.
- × Kiểm tra khoảng cách của các nodes, cập nhật bảng Neighborhood