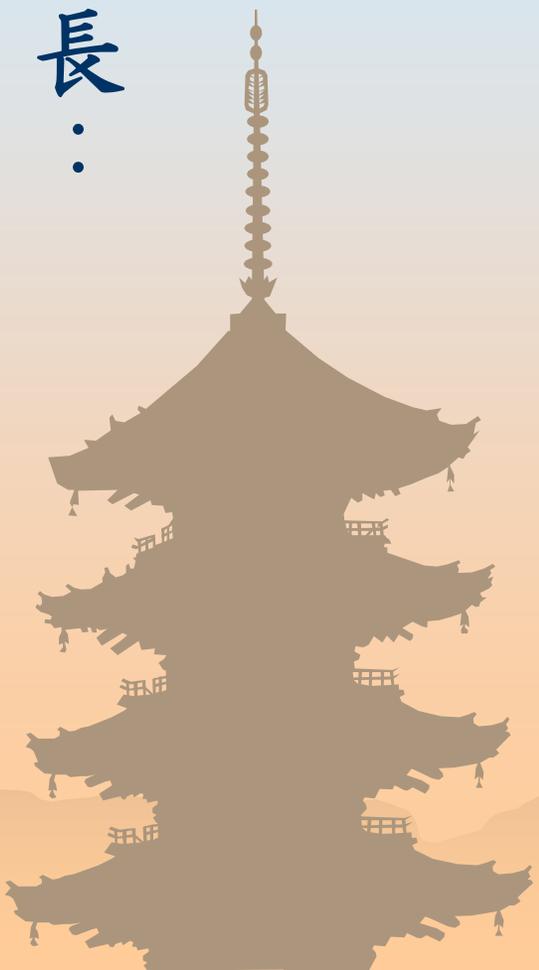


織田信長：

優先度に基づく

並行論理型言語



平田圭二 NTT CS 基礎研究所

山崎憲一 NTT 未来ねっと研究所

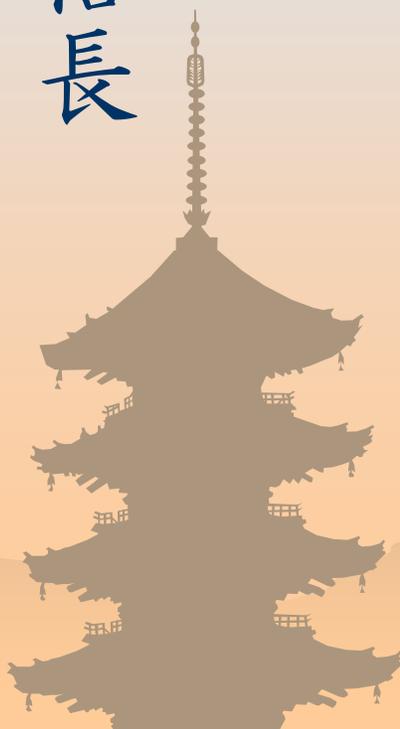
命名織田信長

曰新しい言語

曰新言語

曰信玄後

曰織田信長



本研究の動機

Ⅱ 優先度は不思議

実行順序に何らかの影響を与える
実行順序を直接的に指定しない

Ⅱ でも、優先度が無いと書けない実用的なアプリケーションやアルゴリズムがある

割込み、例外処理、見込み計算、
実時間処理など



本研究の目的

優先度は不思議である

- Ⅱ 宣言的プログラミングに優先度を取り込むこと
- Ⅱ 優先度を形式化すること
- Ⅱ 優先度に基づく並行論理型言語を設計すること



優先度とは何か？

Ⅱ 非決定性の制御

- 確率的

- 決定的

こっち



Ⅱ 二つを比べたら順序は付けられる

Ⅱ 実行に与えた影響は答えに反映される

変数の束縛



FGHC

織田信秀

織田信長

織田家家系図

AND非決定性

$:- X = a, Y = b, p(X,Y), q.$

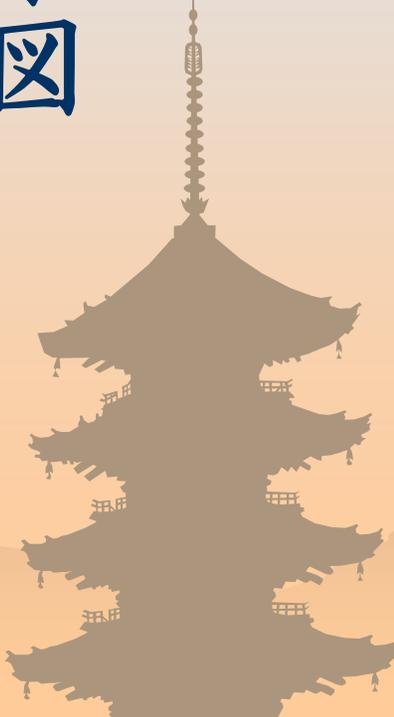
$p(X,Y) :- X = a \mid \text{true}.$

OR非決定性

$p(X,Y) :- Y = b \mid \text{true}.$

$q.$

プロセスとメッセージ



織田信長の設計方針

一 優先度によって二種類の
非決定性を制御する

一 優先度を表現する特別な
データ型

直感的かつ必要最低限

一 優先度を含む項を述語と
して表現する



織田信長の構文

flat な項

全ての述語呼出しと単一化に優先度変数を付与
優先度指定

定義節：

$$p(X_1, \dots, X_n)^\beta :- X_1 \stackrel{\beta_1}{=} t_1, \dots \mid b_1, b_2, \dots$$

述語呼出し： $p(Y_1, \dots, Y_n)^\alpha$

能動単一化： $Y \stackrel{\alpha}{=} t$

優先度指定： $(\alpha_1^i, \dots) > \xi^0 > (\beta_1^i, \dots)$

半順序ではない



織田信長の実行規則

write once,
suspension

1. ボディ部から実行可能な呼出しを選ぶ
各引数と優先度変数が十分具体化されていて
ガード部が成功する

2. 各呼出しの集合優先度を計算する
 $\{\beta, \beta_1, \beta_2, \dots\}$ $\{\alpha\}$ $\{T\}$

プロセスと
メッセージ
同格

非決定的な述語呼出しでは複数の集合優先度が
計算される

3. 極大な集合優先度を持つ呼出しを1つ選ぶ
 $\alpha > \beta$ とすると $\{\alpha\} > \{\alpha, \beta\} > \{\beta\}$

実行規則の補足

II 優先度の伝搬と単一化の関係

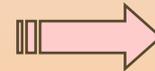
能動単一化: $X_1^{\alpha_1} X_2 \dots X_n^{\alpha_n} = t$

受動単一化: $Y = t^{\beta}$

最小
 $\Rightarrow \beta = \alpha_1 \downarrow \dots \downarrow \alpha_n$

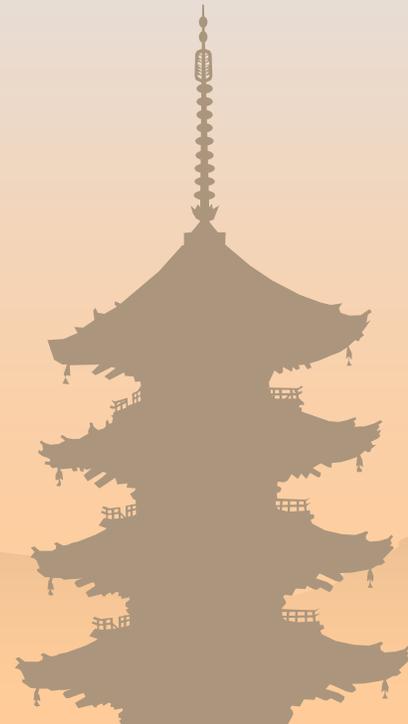
能動単一化: $X \stackrel{\alpha}{=} f(A)$

受動単一化: $Y \stackrel{\beta}{=} f(B)$



$$\beta = \alpha$$

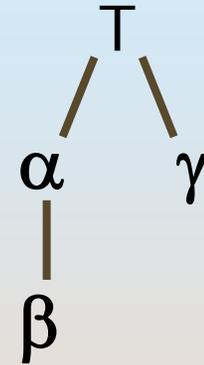
$$A^T = B$$



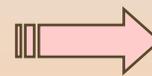
述語呼出しの制御

II プログラム例

$:- T > \alpha^0, \alpha^i > \beta^0, T > \gamma^0,$
 $p(X)^\alpha, q(X)^\beta, X \stackrel{\gamma}{=} t.$



$p(X)^\alpha : - X \stackrel{\gamma}{=} t \mid b_1, \dots$
 $q(X)^\beta : - X \stackrel{\gamma}{=} t \mid b_2, \dots$



$\{\alpha, \gamma\}$
 $\{\beta, \gamma\}$

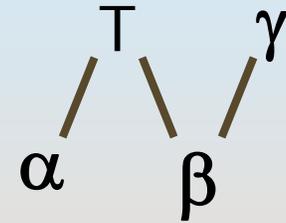
論理変数に関するsuspensionと
優先度変数に関するsuspension



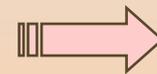
節選択の制御

II プログラム例

$$\begin{aligned} &:- T > \alpha^o, T > \beta^o, \gamma^o > \beta^i, \\ &r(X, Y)^\alpha, X \stackrel{\beta}{=} s, Y \stackrel{\gamma}{=} t. \end{aligned}$$



$$\begin{aligned} r(X, Y)^\alpha &:- X \stackrel{\beta}{=} s \mid b_1, \dots \\ r(X, Y)^\alpha &:- Y \stackrel{\gamma}{=} t \mid b_2, \dots \end{aligned}$$



$\{\alpha, \beta\}$
 $\{\alpha, \gamma\}$

論理変数に関するsuspensionと
優先度変数に関するsuspension

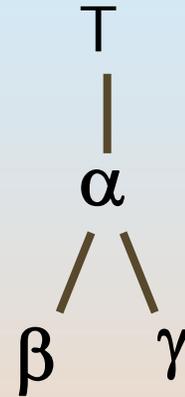


織田信長の怒りをかうプログラム

$$:- T > \alpha^o, \alpha^i > \beta^o, \alpha^i > \gamma^o,$$
$$r(X, Y, \beta, \gamma)^\alpha, X \stackrel{\beta}{=} a, Y \stackrel{\gamma}{=} b.$$

$$r(X, Y, _, \gamma)^\alpha :- X \stackrel{\beta}{=} a \mid \gamma^o > \beta^i.$$

$$r(X, Y, \beta, _)^\alpha :- Y \stackrel{\gamma}{=} b \mid \beta^o > \gamma^i.$$



Ill-modedな優先度変数がある(β, γ)



織田信長の特徴

- 一 優先度は大小関係だけを指定する
- 一 優先度は呼出し側で付与する
- 一 各優先度変数に具体化を行う出現は高々一回である

(well-moded)

- 一 優先度変数が具体化されるまで実行は遅延する



KL1との比較

- II @priority と alternatively
- II 優先度を自然数で表現
- II 呼出し側と定義節側で優先度を付与

矛盾した優先度付けのプログラム例

```
:- (X=a)@priority(10), (Y=b)@priority(20),  
   p(X,Y)@priority(Np), Np: = 数.
```

```
p(X,Y) :- X=a | true.
```

```
alternatively.
```

```
p(X,Y) :- Y=b | true.
```



信長の野望



Ⅱ モデル理論的意味論

Ⅱ 効率的な処理系の作成

スレッド抽出やゴールスケジューリング
優先度計算の省略パターン検出

Ⅱ プログラミング技法

