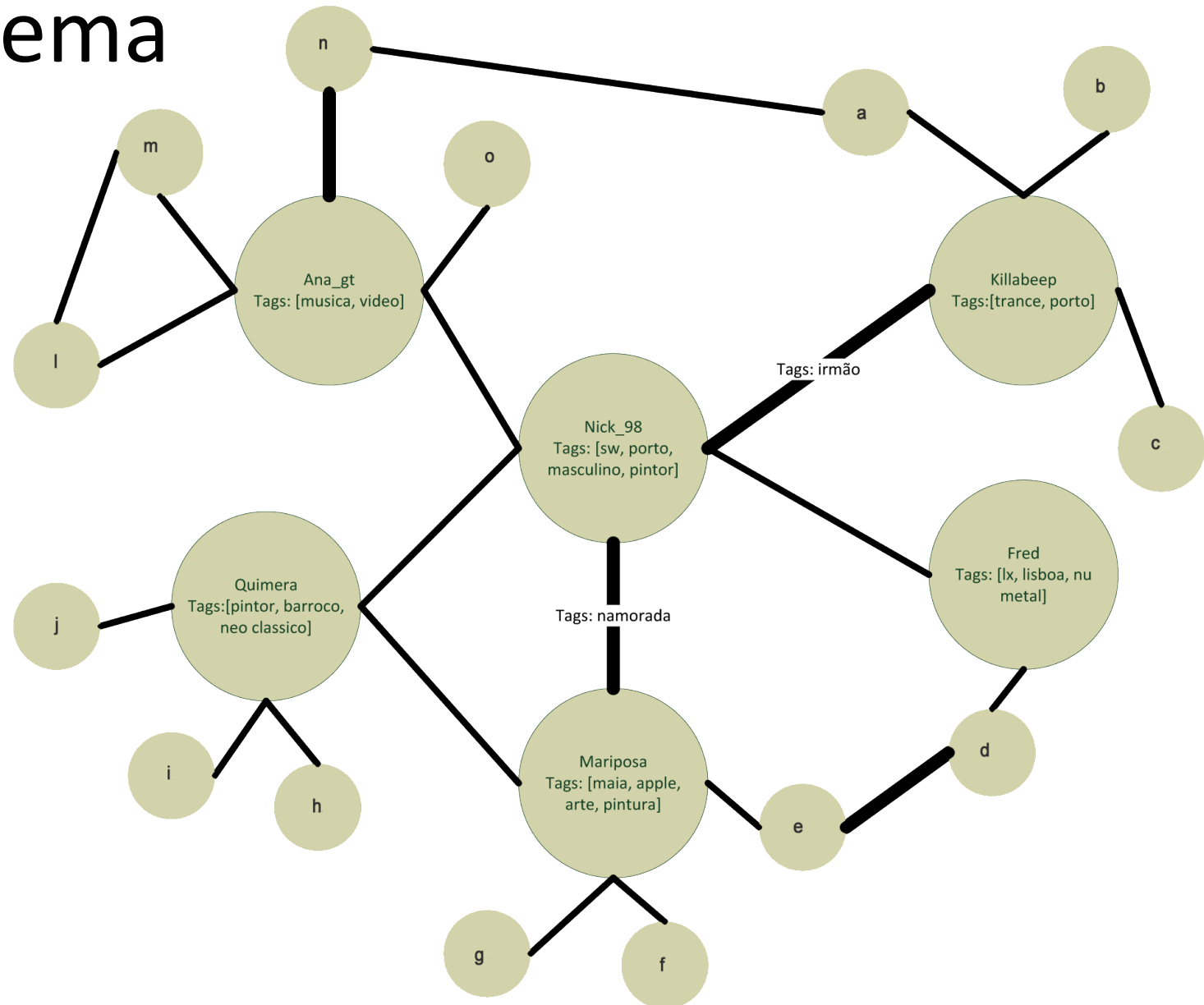


# Social graph visualization

Módulo de Inteligência Artificial de  
LAPR5

# Problema



# Definição das estruturas - exemplo

```
% user_profile(Id, Tags, DataNasc, Tel, Email, Linkedin, Facebook, Estado).  
user_profile(nick_98,[sw,porto,masculino,pintor],_,_,_,_,_,_).  
user_profile(killabeep,[trance,porto],_,_,_,_,_,_).  
user_profile(fred,[lx,lisboa,'nu metal'],_,_,_,_,_,_).  
user_profile(mariposa,[mala,apple,arte,pintura],_,_,_,_,_,_).  
user_profile(quimera,[pintor,barroco,'neo classico'],_,_,_,_,_,_).  
user_profile(ana_gt,[musica,video],_,_,_,_,_,_).
```

LPA PRODATA – permite associar predicados a tabelas de uma BD relacional

Exemplo: relação entre o predicado user\_profile/8 com uma tabela de utilizadores (PDI\_REF.PDF)

Intelligence Server – DLL que implementa comunicação com .NET, por exemplo (INT\_REF.PDF)

# Definição das estruturas - exemplo

```
% user_conn(Id, ConnList).
```

```
% ConnList=[(UserIdA,TagA,Strength1),(UserIdB,TagB,Strength2),...]
```

```
user_conn(killabeep,[(nick_98,irmao,3),(a,_),(b,_),(c,_,_)])
```

```
user_conn(nick_98,[(killabeep,irmao,3),(fred,_),(mariposa,namorada,2),(quimera,_),  
  (ana_gt,_,_)])
```

```
user_conn(fred,[(nick_98,_),(d,_,_)])
```

```
user_conn(mariposa,[(nick_98,namorada,2),(e,_),(f,_),(g,_),(quimera,_,_)])
```

```
user_conn(quimera,[(nick_98,_),(mariposa,_),(h,_),(i,_),(j,_,_)])
```

```
user_conn(ana_gt,[(l,_),(m,_),(n,_),(o,_),(nick_98,_,_)])
```

```
% semantic_eq([tag1, tag2, ..., tagn]).
```

```
semantic_eq(['c#','csharp']).
```

# Objectivos

- Determinar o tamanho da rede de um utilizador (até 3º nível)
- Obter os amigos que tenham em comum X tags sendo X parametrizável. Deve ter em atenção que duas tags sintaticamente diferentes podem ter o mesmo significado semântico (e.g. C# e Csharp)
- Sugerir conexões com outros utilizadores tendo por base as tags e conexões partilhadas (até 3º nível)
- Determinar quem são os “maven” (super conectores) de um determinado assunto (tag). Um vértice estrela ou central do grafo é aquele cujo grau (número de arcos do vértice em causa/ número total de vértices) é mais próximo de 1

# Objetivos

- Determinar Grafo de amigos comuns entre dois utilizadores
- Determinar o caminho mais forte (segundo um peso dado à ligação que reflete a natureza da relação - família, amigo, conhecido, etc.) e o mais curto entre dois utilizadores

$$\min \frac{\sum n_{arcos}}{\sum w_{arcos}}$$

- Grau médio de separação entre utilizadores - para quaisquer dois pares do grafo obter a distância. No caso de grafos de grande dimensão não será possível em tempo útil determinar esta medida, pelo que será necessário recorrer a estimativas. Pretende-se que cada proponente efetue uma análise da complexidade do seu algoritmo e indique o ponto a partir do qual utilizará amostragem e/ou estimativa

$$\bar{d}_G = \frac{1}{n(n-1)} \sum_{i,j} d(v_i, v_j)$$