**Incidence matrix in OOP world**

This talk is devoted to the application of the particular graph representation for the purpose of support of multiple inheritance for methods and fields with different, close to optimal runtime support schemes for dynamic dispatch with possibility of dynamic loading of new classes

1. Introduction and problem statement
   1. Personal introduction
   2. Stereotypes – multiple inheritance is bad, inheritance is evil
   3. History – 1993 FST/CAAST
2. How inheritance graph looks in the incidence matrix form
   1. Key definitions – origin, seed, member version
   2. Example
   3. Sparse matrix – is it an issue?
3. Matrix operations
   1. Rows and columns can be swapped
   2. Numbering algorithm
4. Ideal matrix
   1. Block form matrix
   2. Rows vs. columns (VMT vs. MST)
5. What to do when we load class at runtime?
   1. How to rebuild the matrix?
   2. How to regenerate code?
   3. Smart code generation
6. Summary