```
        ID                  id-at-countryName }

organizationName ATTRIBUTE ::= {
        WITH SYNTAX         DirectoryString {ub-organization-name}
        ID                  id-at-organizationName }

organizationalUnitName ATTRIBUTE ::= {
        WITH SYNTAX         DirectoryString {ub-organizational-unit-name}
        ID                  id-at-organizationalUnitName }

id-at OBJECT IDENTIFIER ::= {joint-iso-ccitt(2) ds(5) 4}

id-at-commonName              OBJECT IDENTIFIER ::= {id-at 3}
id-at-countryName             OBJECT IDENTIFIER ::= {id-at 6}
id-at-organizationName        OBJECT IDENTIFIER ::= {id-at 10}
id-at-organizationalUnitName  OBJECT IDENTIFIER ::= {id-at 11}

DirectoryString { INTEGER:maxSize } ::= CHOICE {
        printableString         PrintableString (SIZE (1..maxSize)),
        utf8String              UTF8String (SIZE(1..maxSize))
}

ub-common-name               INTEGER ::= 64
ub-organization-name         INTEGER ::= 64
ub-organizational-unit-name  INTEGER ::= 64

END
```

２　Implicitly Tagged Module

```
MOJCorpCertImplicit { 1 2 392 100300 1 4 2 }

DEFINITIONS IMPLICIT TAGS ::=
BEGIN

--EXPORTS ALL --

IMPORTS
        Name, CertificateSerialNumber, DirectoryString, EXTENSION
        FROM MOJCorpCertExplicit { 1 2 392 100300 1 4 1 };


authorityKeyIdentifier EXTENSION ::= {
        SYNTAX          AuthorityKeyIdentifier
        IDENTIFIED BY   id-ce-authorityKeyIdentifier }

AuthorityKeyIdentifier ::= SEQUENCE {
    keyIdentifier               [0] KeyIdentifier,
    authorityCertIssuer         [1] GeneralNames,
    authorityCertSerialNumber   [2] CertificateSerialNumber }

KeyIdentifier ::= OCTET STRING

GeneralNames ::= SEQUENCE SIZE (1..MAX) OF GeneralName

GeneralName ::= CHOICE {
        directoryName               [4] Name,
        uniformResourceIdentifier   [6] IA5String
}

subjectKeyIdentifier EXTENSION ::= {
        SYNTAX          SubjectKeyIdentifier
        IDENTIFIED BY   id-ce-subjectKeyIdentifier }

SubjectKeyIdentifier ::= KeyIdentifier

keyUsage EXTENSION ::= {
        SYNTAX KeyUsage
        IDENTIFIED BY id-ce-keyUsage }

KeyUsage ::= BIT STRING {
        digitalSignature     (0),
        nonRepudiation       (1),
        keyEncipherment      (2),
        dataEncipherment     (3),
        keyAgreement         (4),
        keyCertSign          (5),
        cRLSign              (6)
        encipherOnly         (7),
        decipherOnly         (8)
}

privateKeyUsagePeriod EXTENSION ::= {
        SYNTAX  PrivateKeyUsagePeriod
        IDENTIFIED BY { id-ce-privateKeyUsagePeriod } }

PrivateKeyUsagePeriod ::= SEQUENCE {
        notBefore       [0]     GeneralizedTime,
        notAfter        [1]     GeneralizedTime }

certificatePolicies EXTENSION ::= {
        SYNTAX  CertificatePoliciesSyntax
```