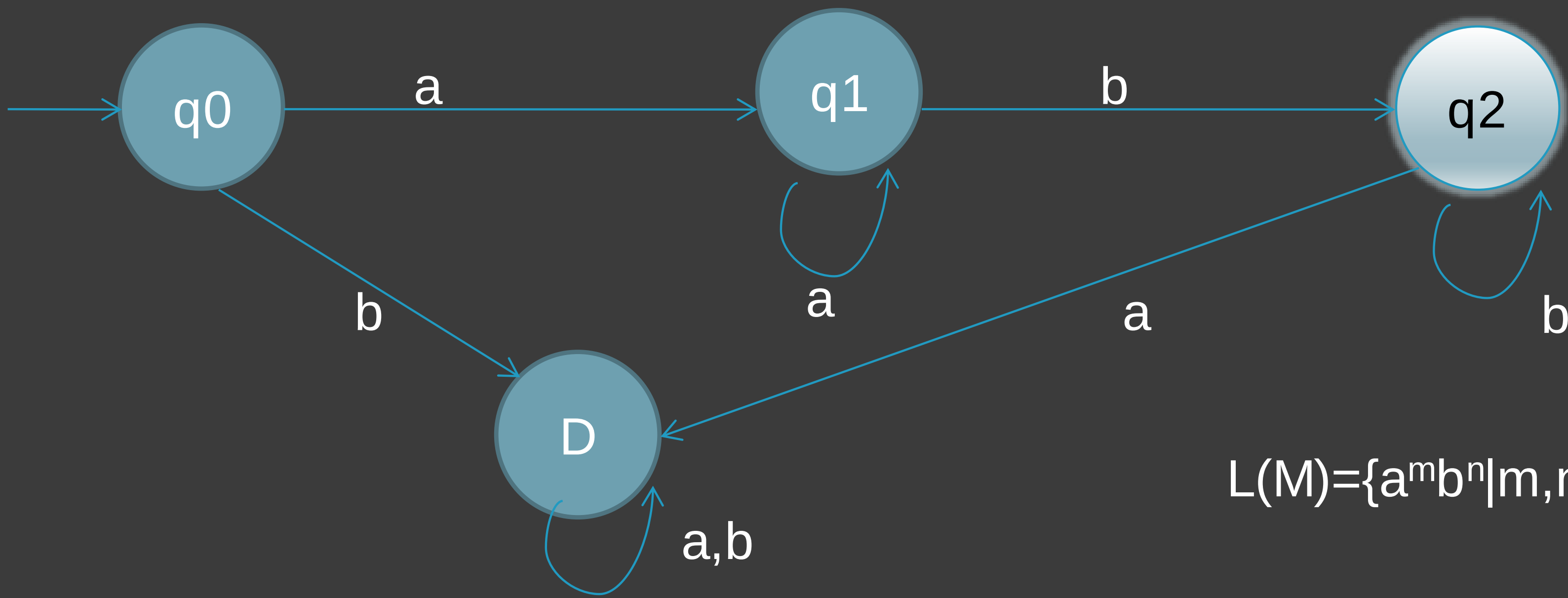
Conversion of FSA(DFA) to RG

- Given DFA $M=(Q,\sum,\delta,q_0,F)$ construct RG $G=(N,T,P,S)$
- Such that $L(G)=L(M)$ or $L(M)=L(G)$
- Construction
- $N=Q$
- $T=\sum$
- $S=q_0$
- Now definition of P
- 1. If $\delta(q,a)=p$ then $q \to ap$ is a rule in P
- 2. If $\delta(q,a)=p$ and $p \varepsilon F$ then $q \to a$ is rule in P

Conversion of DFA to RG



L(M)={$a^m b^n$|m,n>=1}

DFA M ; RG G

1. δ(q0,a)=q1 ; q0->aq1
2. δ(q0,b)=D ; q0->bD
3. δ(q1,a)=q1 ; q1->aq1
4. δ(q1,b)=q2 ; q1->bq2 and q1->b as q2 is final state
5. δ(q2,a)=D ; q2->aD
6. δ(q2,b)=q2 ; q2->bq2 and q2->b as q2 is final state
7. δ(D,a)=D ; D->aD
7. δ(D,b)=D ; D->bD

L(G)={$a^m b^n$|m,n>=1}

Regular Language/ Regular Set

- Regular Grammar generates Regular Language
- Finite State Automata accepts Regular Language
- Regular Expression represents Regular Language
- **RG≡FSA≡RE [RL]**